N71-32793

NASA TECHNICAL MEMORANDUM

NASA TM X-2343

NASA TM X-2343

COMPUTER PROGRAM FOR DESIGN OF TWO-DIMENSIONAL SHARP-EDGED-THROAT SUPERSONIC NOZZLE WITH BOUNDARY-LAYER CORRECTION

by Louis J. Goldman and Michael R. Vanco

Lewis Research Center

Cleveland, Ohio 44135

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION • WASHINGTON, D. C. • AUGUST 1971

1. Report No. NASA TM X-2343	2. Government Accession No.	3. Recipient's Catalog No.
4. Title and Subtitle COMPUTER PROGRAM FOR DESIGN OF TWO-DIMENSIONAL SHARP-EDGED-THROAT SUPERSONIC NOZZLE WITH BOUNDARY-LAYER CORRECTION		5. Report Date August 1971
		6. Performing Organization Code
7. Author(s) Louis J. Goldman and Michael R. Vanco		8. Performing Organization Report No. E-6261
		10. Work Unit No. 120-34
9. Performing Organization Name and Address Lewis Research Center National Aeronautics and Space Administration Cleveland, Ohio 44135		11. Contract or Grant No.
		13. Type of Report and Period Covered Technical Memorandum
12. Sponsoring Agency Name and Address National Aeronautics and Space Administration Washington, D.C. 20546		14. Sponsoring Agency Code
15. Supplementary Notes		

16. Abstract

A FORTRAN IV computer program is presented for the design of sharp-edged-throat supersonic nozzles where losses are accounted for by correcting the ideal nozzle geometry for boundary-layer displacement thickness. The ideal nozzle is designed by the method of characteristics. Boundary-layer parameters are calculated by Cohen and Reshotko's method for laminar flow and Sasman and Cresci's method for turbulent flow.

17. Key Words (Suggested by Author(s))	18. Distribution Statement
Supersonic nozzle Nozzle Supersonic flow	Unclassified - unlimited

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price*
Unclassified	Unclassified	61	$3.00

*For sale by the National Technical Information Service, Springfield, Virginia 22151

COMPUTER PROGRAM FOR DESIGN OF TWO-DIMENSIONAL SHARP-EDGED-THROAT SUPERSONIC NOZZLE WITH BOUNDARY-LAYER CORRECTION

by Louis J. Goldman and Michael R. Vanco

Lewis Research Center

SUMMARY

A FORTRAN IV computer program is presented for the design of sharp-edged-throat supersonic nozzles where losses are accounted for by correcting the ideal nozzle geometry for boundary-layer displacement thickness. The ideal nozzle is designed by the method of characteristics. Boundary-layer parameters are calculated by Cohen and Reshotko's method for laminar flow and by Sasman and Cresci's method for turbulent flow. The final nozzle geometry is then obtained by adding the displacement thicknesses to the ideal nozzle coordinates. The program is essentially a modified combination of two programs presented previously.

The computer program input consists primarily of the exit Mach number, specific heat ratio, and total flow conditions. The program output gives the corrected nozzle geometry. An example of the output is included to indicate the use of the program.

INTRODUCTION

Methods for the design of supersonic turbines for possible use in turbopump and open-cycle auxiliary power systems where high-energy fluids are used and high pressure ratios are available have recently become of interest. The method of characteristics as applied to the two-dimensional isentropic flow of a perfect gas can be used for the design of the supersonic blading. Computer programs for the isentropic design of two-dimensional supersonic nozzles and rotor blade sections have been described in reference 1 and 2, respectively. The design of blading by these procedures must then be supplemented by a knowledge of the losses that occur in the nozzle and rotor.

The purpose of this report is to present a computer program for the design of sharp-edged-throat supersonic nozzles where losses are accounted for by correcting the ideal nozzle geometry for boundary-layer displacement thickness. The ideal nozzle

geometry is obtained using the computer program described in reference 1. Boundary-layer parameters are calculated using the computer program described in reference 3. The final nozzle geometry is then obtained by adding the displacement thicknesses to the ideal nozzle coordinates. The program described herein is essentially a modified combination of the two programs presented in references 1 and 3.

The boundary-layer parameters are also used to calculate the conditions downstream of the nozzle after the flow has mixed to a uniform state. The procedure described in reference 4 is used for this purpose.

In this report, a description of the input and output and a complete FORTRAN IV listing of the program are presented. A brief description of the computer program and method of design are also given. An example of the output is included to indicate the use of the program.

METHOD OF ANALYSIS

The design of sharp-edged-throat supersonic nozzles where losses are accounted for by correcting the ideal nozzle geometry for boundary-layer displacement thickness is described herein. This type of ideal nozzle is desirable since it produces uniform parallel flow at the exit in the smallest possible distance. The ideal symmetric nozzle is designed by the method of characteristics as applied to the isentropic flow of a perfect gas. The boundary-layer parameters (displacement and momentum thicknesses) are then calculated for the ideal nozzles. The final nozzle geometry is obtained by adding the displacement thicknesses to the ideal nozzle coordinates. In addition, the boundary-layer parameters are used to calculate the aftermixing conditions downstream of the nozzle assuming the flow mixes to a uniform state.

Nozzle Description

A typical supersonic turbine nozzle, as seen in figure 1, consists of three sections: (1) a converging (subsonic) section, (2) a diverging (supersonic) section, and (3) a straight section on the suction surface. The converging section produces the flow turning with little losses and is not designed by the program. The diverging section accelerates the flow to the desired free-stream Mach number at the exit. This section is designed by the method of characteristics using the program described in reference 1. The straight section on the suction surface completes the nozzle profile, and its length is determined by the required nozzle angle (see fig. 1).

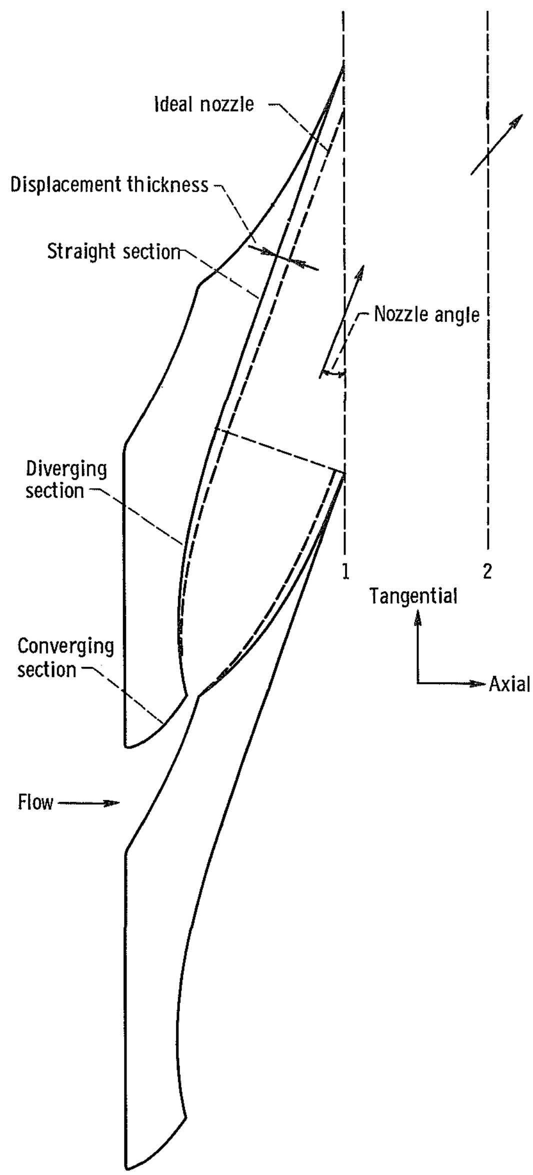

Figure 1. - Design of supersonic nozzle with sharp-edged throat.

Boundary-Layer Calculations

The boundary-layer parameters (displacement and momentum thicknesses) are calculated for the ideal nozzle using the computer program described in reference 3. The program uses Cohen and Reshotko's method (ref. 5) for the calculation of laminar boundary layers and Sasman and Cresci's method (ref. 6) for turbulent boundary layers. Curvature effects are not considered in these calculations.

In the laminar regime, a single ordinary differential equation (the momentum integral equation) is solved numerically. As explained in reference 5, the results have to be extended for flows in highly favorable pressure gradients as would occur in a nozzle. For turbulent flow, coupled first-order ordinary differential equations (the momentum and moment-of-momentum integral equations) are solved using Runge-Kutta techniques. Transition from laminar to turbulent flow, if it occurs, is predicted by the program.

The displacement thicknesses obtained from the program are then added to the ideal nozzle coordinates to obtain the final nozzle profile. Figure 1 shows a nozzle designed in this manner. The dashed line represents the ideal nozzle profile.

Aftermixing Conditions

The boundary-layer parameters (displacement and momentum thickness) at the nozzle exit are also used to calculate the aftermixing conditions by the method given by Stewart in reference 4. In this loss model, the flow sufficiently downstream of the blade row is assumed to be mixed to a uniform condition. Application of the continuity, momentum, and energy equations between stations 1 and 2 (fig. 1) results in the determination of the aftermixing conditions.

Subsonic and supersonic aftermixing axial Mach number solutions are possible for this loss model when the free-stream axial Mach number at the nozzle exit (before mixing) is supersonic. The subsonic solution corresponds to mixing plus oblique shock losses, whereas the supersonic solution corresponds to shockless mixing. A more detailed discussion of the different solutions can be found in reference 7.

DESCRIPTION OF INPUT

A description of the input for the FORTRAN IV computer program is given in this section. The input quantities consist essentially of the nozzle-exit Mach number, specific-heat ratio, nozzle angle, throat half-height, nozzle subsonic section coordinates and corresponding pressure ratios, total temperature, total pressure, gas constant, and initial momentum or displacement thickness. The program gas properties are set

up for air. For gases other than air the changes required to the program are as described in appendix A.

The input format is shown in table I. The input variables are the following:

ME	nozzle-exit Mach number
DV	increment in flow turning (recommended value, 0.1), deg
GAM	specific-heat ratio
NP	nozzle coordinate printout indicator for supersonic section (e.g., if NP = 10, every tenth coordinate is printed out; it should also be noted that for any value of NP, the last coordinate is always printed out, and NP must be an integer and right adjusted)
NS	number of input points for nozzle inlet including the throat (must have at least one point, the throat point)
ALP1	nozzle angle measured from tangential direction, deg, see fig. 1
YTH	throat half-height, m (ft)
XCP	array of X-coordinates of nozzle subsonic section (based on throat half-height of unity)
YCP	array of Y-coordinates of nozzle subsonic section (based on throat half-height of unity)
PSPT	array of static to total pressure ratios corresponding to the nozzle subsonic section coordinates XCP and YCP
R	gas constant, J/(kg)(K) ((ft)(lbf)/(slug)(^{o}R))
PTZ	inlet or upstream total pressure, N/m^2 (lbf/ft^2)
TTZ	inlet or upstream total temperature, K (^{o}R)
UPMACH	inlet or upstream Mach number
NVP	integer number of points desired in velocity profile of boundary layer at each station (must have at least 1)
NTURB	integer number of station, if any, at which user wishes turbulent boundary layer to begin (NTURB is usually zero, allowing the program to calculate position of transition to turbulent boundary layer; if NTURB = 1, initial values must be given for DTURB and TTURB)

KEM integer (0 or 1) indicating which of two allowable sets of units are used in input:

U.S. Customary (lbf, slugs, ft, sec, $^{\circ}$R, and ft-lb) 0

International System (N, kg, m, sec, K, and J) 1

DLAM initial displacement thickness, if any, of laminar boundary layer at first point, m (ft) (DLAM may be zero, the stagnation point, or have some finite value)

TLAM initial momentum thickness, if any, of laminar boundary layer at first point, m (ft) (TLAM may be zero, or have some finite value)

DTURB initial displacement thickness, if any, of turbulent boundary layer, m (ft)

TTURB initial momentum thickness, if any, of turbulent boundary layer, m (ft)

KPRE integer (0 or 1) indicating whether printing of output from PRECAL is desired:

Output suppressed 0

Output printed 1

KGRAD integer (0 or 1, see KPRE) indicating whether printing of surface gradients of velocity and Mach number is desired

KSDE integer (0 or 1, see KPRE) indicating whether printing of solutions of laminar and turbulent differential equations is desired

KLAM integer (0 or 1, see KPRE) indicating whether printing of laminar calculations for location of instability and transition is desired

KMAIN integer (0 or 1, see KPRE) indicating whether printing of principal calculated boundary-layer parameters is desired

KPROF integer (0 or 1, see KPRE) indicating whether printing of velocity profiles is desired

TE thickness of trailing edge, m (ft)

TABLE I. - INPUT FORM

[Numbers in corners are card column numbers.]

1 6 ME	7 12 DV	13 18 GAM	20 NP	22 NS	23 28 ALP1	29 38 YTH	

10 XCP	20 YCP	30 PSPT	40

10 R	20 PTZ	30 TTZ	40 UPMACH

5 NVP	10 NTURB	15 KEM	

10 DLAM	20 TLAM	30 DTURB	40 TTURB

5 KPRE	10 KGRAD	15 KSDE	20 KLAM	25 KMAIN	30 KPROF	

6 TE	

SPECIAL INSTRUCTIONS FOR PREPARING INPUT

Subsonic Section

For this program, the subsonic section of the nozzle is used to determine the boundary-layer parameters at the throat. If the displacement thickness or momentum thickness at the throat is known, it is put in as input, DLAM or TLAM. Then, only one point is required for the subsonic section, the throat point. At this point, $X = 0$, $Y = 1.0$, and PSPT equals the static to total pressure ratio at Mach 1.

If the boundary-layer parameters are not known, a subsonic section is put in. Fig-

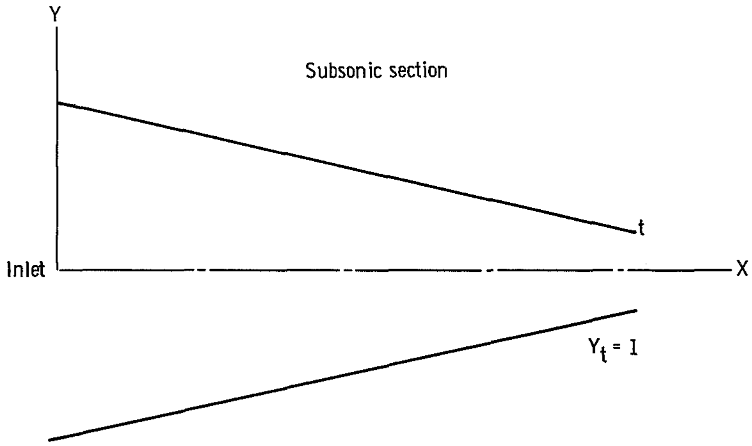

Figure 2. - Coordinate system for geometry input. All coordinates referenced to throat half-height of unity.

ure 2 shows an example subsonic section with the required coordinate system. The subsonic section must be symmetrical because only one surface is used, and the X-Y coordinates must be referenced to a throat half-height of unity. Also, along with coordinates, the static to total pressure ratio (PSPT) at each point is needed.

Output

Usually KPRE, KGRAD, KSDE, KLAM, and KPROF are set equal to zero. They do not give the main output of the boundary-layer section. However, if this additional output is desired, these quantities are set equal to 1. A description of this output is given in appendix B.

DESCRIPTION OF MAIN OUTPUT

An example of the output from the program is shown in table II. This output is in U.S. Customary units. Each section of the output has been numbered to correspond to the following description:

(1) The first output of the program is the input values specified for the design of the ideal (or loss-free) supersonic nozzle except for VE, the Prandtl-Meyer angle corresponding to the exit Mach number, ME. This output, also, contains the nozzle subsonic section coordinates.

(2) Output 2 gives a table of the coordinates (X and Y) of the supersonic portion of the ideal nozzle based on a throat half-height of unity together with the corresponding Mach number (M) and total to static pressure ratio (PTPS).

(3) Output 3 gives the input to the boundary-layer program. The first three lines of this output are a listing of the input data except for NST, which is the total number of points. The remaining output is the nozzle coordinates for the suction surface which contains 10 points on the straight section:

X X-coordinate, m (ft)

Y Y-coordinate, m (ft)

POPTZ static to total pressure ratio corresponding to X and Y

TWAL wall temperature, K (^{o}R) (wall temperatures are assumed equal to total temperature, TTZ)

(4) Output 4 lists the properties calculated for the upstream conditions:

PSZ upstream static pressure, N/m^2 (lb/ft^2)

TSZ upstream static temperature, K (^{o}R)

UZ upstream velocity, m/sec (ft/sec)

ASZ (ATZ) speed of sound based on upstream static (total) temperature, m/sec (ft/sec)

RHSZ (RHTZ) static (total) density based on upstream static (total) temperature, kg/m^3 (slug/ft^3)

MUSZ (MUTZ) dynamic viscosity based on upstream static (total) temperature, (N)(sec)/m^2 ((lbf)(sec)/ft^2)

NUSZ (NUTZ) kinematic viscosity based on upstream static (total) temperature, m^2/sec (ft^2/sec)

CP specific heat at constant pressure, J/(kg)(K) ((ft)(lbf)/(slug)(^{o}R))

PR Prandtl number

TC thermal conductivity, J/(m)(sec)(K) ((ft)(lbf)/(ft)(sec)(^{o}R))

ARCL total distance along surface in x-direction, m (ft)

The next part of this output gives the variables describing the flow along the surface:

PRES static pressure, N/m^2 (lbf/ft^2)

UE free-stream velocity, m/sec (ft/sec)

ME free-stream Mach number

POPTZ static to total pressure ratio

VOVCR ratio of the velocity to the critical velocity

(5) Output 5 corresponds to KMAIN. It indicates the regions of laminar and turbu-

lent boundary layers, and the stations at which instability, transition, and separation occur. It gives all the principal boundary-layer output parameters:

X	X-coordinate, m (ft)
S	surface length, m (ft)
DELSR	displacement thickness, m (ft)
THET	momentum thickness, m (ft)
DELTA	boundary-layer thickness, m (ft)
FORM	compressible form factor
FORMI	incompressible form factor

The next part of this output gives the skin-friction and heat-transfer parameters:

CF	skin-friction coefficient at the wall
TAUW	shear stress at the wall, N/m^2 (lbf/ft^2)
RTH	momentum-thickness Reynolds number
DTDY	slope of temperature profile at the wall, K/m ($^oR/ft$)
NUSS	local Nusselt number
HTRAN	heat transfer per unit area, $J/(sec)(m^2)$ $((ft)(lbf)/(sec)(ft^2))$
CRN	Reynolds analogy parameter

(6) Output 6 lists the aftermixing properties. Variable names ending in 0 refer to the upstream station, those ending in 1 refer to the nozzle exit (before mixing), and those ending in 2 refer to the mixed conditions downstream of the nozzle. The output consists of nozzle exit free-stream Mach number (XMFS1), nozzle exit spacing (SPACING) in meters (ft), trailing-edge thickness (TE) in meters (ft), Mach numbers (XM1 and XM2), axial Mach numbers (XMX1 and XMX2), critical velocity ratios (V/VCR1 and V/VCR2), flow angles measured from axial direction (ALPH1 and ALPH2) in degrees, total to total pressure (PT2/PT0), total to static pressure ratio (PT0/P2), static to total temperature ratio (T2/TT0), nozzle kinetic energy loss coefficient (EBAR2), and nozzle efficiency (ETA-N). For the case with boundary-layer correction, the displacement thickness (DELSR) and the momentum thickness (THET) on the suction surface calculated in AFMIX are printed out. The (N+2) values were used in calculating the aftermixing properties. These values were calculated because the length of the straight section on the suction surface increases when the boundary-layer correction is added and a constant nozzle angle is desired.

(7) The final output 7 lists the coordinates of the supersonic portion of the nozzle corrected for the boundary-layer displacement thickness. The values of the nozzle coordinates (X and Y) are with the nozzle axis axial, while XTU, YTU, XTL, and YTL are the coordinates of suction and pressure surfaces, respectively, with the axis at the nozzle flow angle (see fig. 1).

TABLE II. - SAMPLE OUTPUT

TWO DIMENSIONAL SUPERSONIC NOZZLE WITH A SHARP-EDGED THROAT

VE = 19.999   ME = 1.770   GAMMA = 1.390   DELTA V =0.10000   YTH = 0.11000E-01   ALP1 = 20.000

SUBSCNIC SECTION (INPUT)

	X	Y	PSPT
	0.	5.82180	0.99303
	10.19990	4.18240	0.98640
	14.67270	3.46350	0.98003
	17.78360	2.96350	0.97250
	20.06510	2.59680	0.96383
1	22.29440	2.23850	0.95060
	23.55990	2.03510	0.93947
	25.42200	1.73580	0.91433
	26.32850	1.59010	0.89562
	27.20830	1.44870	0.87027
	27.88580	1.33980	0.84302
	28.96970	1.16560	0.77164
	29.57630	1.06810	0.69507
	29.87120	1.02070	0.62351
	30.00000	1.00000	0.52828

SUPERSCNIC SECTION

	X	Y	M	PTPS
	0.	1.00000	1.00000	1.8868924
	1.12958	1.19752	1.44655	3.3859348
	1.34182	1.23233	1.46352	3.4692084
	1.50189	1.25713	1.48044	3.5546543
	1.64083	1.27741	1.49731	3.6423670
	1.76862	1.29492	1.51414	3.7324436
	1.88990	1.31046	1.53095	3.8249831
	2.00713	1.32444	1.54773	3.9200864
	2.12195	1.33712	1.56450	4.0178579
2	2.23541	1.34864	1.58126	4.1184044
	2.34826	1.35911	1.59801	4.2218362
	2.46108	1.36859	1.61477	4.3282676
	2.57430	1.37710	1.63154	4.4378166
	2.68829	1.38467	1.64833	4.5506049
	2.80334	1.39130	1.66513	4.6667590
	2.91972	1.39699	1.68196	4.7864097
	3.03764	1.40173	1.69881	4.9096938
	3.15732	1.40548	1.71570	5.0367521
	3.27894	1.40824	1.73262	5.1677312
	3.40266	1.40997	1.74958	5.3027837
	3.52867	1.41063	1.76659	5.4420699
	3.52867	1.41063	1.76659	5.4420699

TABLE II. - Continued. SAMPLE OUTPUT

BOUNDARY LAYER SECTION

3

R	PTZ	TTZ	UPMACH		
1716.00	12696.00	1010.00	0.		
NST	NVP	NTURB	KEM		
45	5	-0	-0		
BLAM	TLAM	DTURB	TTURB		
0.000250	0.000100	-0.	-0.		
KPRE	KGRAD	KSDE	KLAM	KMAIN	KPROF
-0	-0	-0	-0	1	-0

X	Y	POPTZ	TWAL
0.	0.06404	0.993030	1010.0000
0.11220	0.04601	0.986400	1010.0000
0.16140	0.03810	0.980030	1010.0000
0.19562	0.03260	0.972500	1010.0000
0.22072	0.02856	0.963830	1010.0000
0.24524	0.02462	0.950600	1010.0000
0.25916	0.02239	0.939470	1010.0000
0.27964	0.01909	0.914330	1010.0000
0.28961	0.01749	0.895620	1010.0000
0.29929	0.01594	0.870270	1010.0000
0.30674	0.01474	0.843020	1010.0000
0.31867	0.01282	0.771640	1010.0000
0.32534	0.01175	0.695070	1010.0000
0.32858	0.01123	0.623510	1010.0000
0.33000	0.01100	0.529972	1010.0000
0.34243	0.01317	0.295339	1010.0000
0.34476	0.01356	0.288250	1010.0000
0.34652	0.01383	0.281321	1010.0000
0.34805	0.01405	0.274547	1010.0000
0.34945	0.01424	0.267921	1010.0000
0.35079	0.01442	0.261439	1010.0000
0.35208	0.01457	0.255096	1010.0000
0.35334	0.01471	0.248889	1010.0000
0.35459	0.01484	0.242812	1010.0000
0.35583	0.01495	0.236864	1010.0000
0.35707	0.01505	0.231039	1010.0000

PRELIMINARY CALCULATIONS

4

```
PSZ    =   12696.00000
TSZ    =   1010.0000
UZ     =      0.

ASZ    =    1552.1251
ATZ    =    1552.1251

RHSZ   =    0.7325348E-02
RHTZ   =    0.7325348E-02

MUSZ   =    0.6048975E-06
MUTZ   =    0.6048975E-06

NUSZ   =    0.8257594E-04
NUTZ   =    0.8257594E-04

CP     =    6115.9999
PR     =    0.66353
TC     =    0.5663117E-02
ARCL   =    0.4586
```

4

STATION	PRES	UE	ME	POPTZ	VOVCR
1	12607.50879	155.63106	0.100368	0.993030	0.109611
2	12523.33435	217.65661	0.140501	0.986400	0.153295
3	12442.46082	264.05568	0.170607	0.980030	0.185974
4	12346.85986	310.29330	0.200699	0.972500	0.218539
5	12236.78564	356.43050	0.230830	0.963830	0.251034
6	12068.81750	417.57410	0.270953	0.950600	0.294097
7	11927.51111	463.19551	0.301052	0.939470	0.326228
8	11608.33362	553.70406	0.361250	0.914330	0.389973
9	11370.79150	613.41231	0.401368	0.895620	0.432026
10	11048.94788	687.30137	0.451530	0.870270	0.484066
11	10702.98181	760.24225	0.501683	0.843020	0.535438
12	9796.74146	930.96562	0.622016	0.771640	0.655678
13	8824.60864	1094.82568	0.742300	0.695070	0.771085
14	7916.08295	1238.37828	0.852526	0.623510	0.872189
15	6728.52344	1419.85098	1.000000	0.529972	1.000000
16	3749.62912	1892.13937	1.446551	0.295339	1.332632
17	3659.62442	1907.82626	1.463524	0.288250	1.343681

TABLE II. - Continued. SAMPLE OUTPUT

PRINCIPAL BOUNDARY LAYER INFORMATION

INSTABILITY DOES NOT OCCUR

TRANSITION DOES NOT OCCUR

5 SEPARATION DOES NOT OCCUR

LAMINAR BOUNDARY LAYER - STATIONS 1 TO 45

TURBULENT BOUNDARY LAYER DOES NOT OCCUR

5

STATION	X	S	DELSR	THET	DELTA	FORM	FORMI
1	0.	0.	0.000243	0.000100	0.000832	2.4340	2.4276
2	0.112199	0.113639	0.000250	0.000100	0.000921	2.5054	2.4927
3	0.161400	0.163471	0.000232	0.000092	0.000886	2.5356	2.5167
4	0.195620	0.198130	0.000209	0.000082	0.000815	2.5593	2.5330
5	0.220716	0.223549	0.000188	0.000073	0.000740	2.5781	2.5432
6	0.245238	0.248386	0.000166	0.000064	0.000656	2.6032	2.5549
7	0.259159	0.262485	0.000152	0.000058	0.000602	2.6209	2.5613
8	0.279642	0.283231	0.000131	0.000049	0.000520	2.6641	2.5778
9	0.289613	0.293330	0.000120	0.000045	0.000474	2.6907	2.5839
10	0.299291	0.303132	0.000110	0.000040	0.000441	2.7452	2.6090
11	0.306744	0.310681	0.000100	0.000036	0.000385	2.7623	2.5950
12	0.318667	0.322756	0.000089	0.000030	0.000380	2.9886	2.7218
13	0.325339	0.329515	0.000072	0.000024	0.000244	2.9676	2.6006
14	0.328583	0.332800	0.000077	0.000021	0.000412	3.6684	3.1119
15	0.330000	0.334235	0.000068	0.000018	0.000314	3.7926	3.0408
16	0.342425	0.346849	0.000089	0.000022	0.000210	4.0912	2.6696
17	0.344760	0.349215	0.000096	0.000023	0.000230	4.1602	2.6946
18	0.346521	0.350997	0.000102	0.000024	0.000245	4.2218	2.7138
19	0.348049	0.352541	0.000107	0.000025	0.000257	4.2789	2.7295
20	0.349455	0.353960	0.000111	0.000026	0.000268	4.3328	2.7426
21	0.350789	0.355305	0.000116	0.000026	0.000279	4.3848	2.7539
22	0.352078	0.356604	0.000120	0.000027	0.000289	4.4353	2.7638
23	0.353341	0.357874	0.000124	0.000028	0.000298	4.4848	2.7727
24	0.354590	0.359129	0.000128	0.000028	0.000307	4.5337	2.7807
25	0.355831	0.360376	0.000132	0.000029	0.000316	4.5822	2.7882
26	0.357072	0.361621	0.000135	0.000029	0.000324	4.6304	2.7951
27	0.358317	0.362870	0.000139	0.000030	0.000333	4.6786	2.8015

5

STATION	CF	TAUW	RTH	DTDY	NUSS	HTRAN	CRN
1	0.	0.	187.2	0.	0.	0.	2.799
2	0.00287	0.49111	259.9	-978.37	154.81	-5.5407	4.842
3	0.00277	0.69284	287.2	-1340.26	207.28	-7.5900	6.050
4	0.00276	0.94582	298.9	-1911.08	259.42	-10.8227	6.806
5	0.00276	1.23823	304.0	-2708.88	314.43	-15.3407	7.221
6	0.00279	1.69433	305.9	-4098.73	385.15	-23.2116	7.650
7	0.00281	2.07543	305.2	-5445.08	439.44	-30.8361	7.824
8	0.00290	2.97685	300.7	-8772.62	534.61	-49.6803	8.327
9	0.00293	3.61860	296.3	-11754.33	604.46	-66.5661	8.369
10	0.00311	4.67552	289.5	-15313.05	648.22	-86.7196	9.300
11	0.00300	5.35578	281.7	-22139.43	785.05	-125.3782	8.151
12	0.00399	9.77562	259.1	-29728.16	730.29	-168.3540	13.567
13	0.00322	9.81642	224.3	-78886.76	1430.57	-446.7449	6.038
14	0.00809	28.34312	197.3	-50701.23	725.80	-287.1270	30.681
15	0.00817	31.98068	162.6	-80457.29	879.94	-455.6390	25.012
16	0.00259	10.03554	146.8	-332898.43	2127.49	-1885.2427	2.528
17	0.00268	10.28577	154.4	-298177.55	1887.17	-1688.6143	2.917
18	0.00274	10.45993	158.5	-274410.61	1717.63	-1554.0193	3.249
19	0.00280	10.57238	161.2	-256461.62	1587.04	-1452.3721	3.542
20	0.00284	10.63108	163.0	-242547.11	1483.79	-1373.5726	3.795
21	0.00287	10.65249	164.3	-231373.06	1399.37	-1310.2926	4.017
22	0.00290	10.64292	165.3	-222221.00	1328.97	-1258.4635	4.210
23	0.00291	10.60644	166.0	-214596.34	1269.22	-1215.2841	4.376
24	0.00293	10.54865	166.5	-208095.97	1217.46	-1178.4717	4.521
25	0.00294	10.47273	166.8	-202460.90	1171.94	-1146.5597	4.646
26	0.00295	10.38160	167.1	-197493.29	1131.32	-1118.4275	4.754
27	0.00295	10.27764	167.2	-193047.90	1094.63	-1093.2528	4.848
28	0.00295	10.16284	167.2	-189014.53	1061.12	-1070.4113	4.929
29	0.00295	10.03898	167.1	-185306.65	1030.23	-1049.4132	4.999
30	0.00295	9.90755	167.0	-181858.09	1001.49	-1029.8836	5.060
31	0.00294	9.76977	166.9	-178617.79	974.57	-1011.5334	5.113
32	0.00291	9.54961	166.7	-172338.05	931.84	-975.9705	5.213
33	0.00291	9.40499	166.5	-168505.05	903.13	-954.2638	5.283
34	0.00286	9.13631	166.2	-173826.39	923.69	-984.3992	5.006
35	0.00170	5.35489	165.9	-250520.49	1320.16	-1418.7268	2.048
36	0.00144	4.53268	184.1	-227632.28	1226.94	-1289.1082	1.908
37	0.00138	4.34516	202.1	-206044.66	1135.38	-1166.8550	2.021
38	0.00128	4.01812	218.5	-190536.64	1072.85	-1079.0312	2.021
39	0.00119	3.75531	233.8	-178074.28	1024.11	-1008.4554	2.021
40	0.00112	3.53813	248.2	-167775.60	985.08	-950.1328	2.021
41	0.00107	3.35474	261.7	-159079.22	953.16	-900.8842	2.021
42	0.00102	3.19719	274.6	-151608.55	926.64	-858.5769	2.021
43	0.00097	3.05994	286.9	-145100.33	904.33	-821.7201	2.021
44	0.00093	2.93898	298.8	-139364.19	885.35	-789.2356	2.021
45	0.00090	2.83131	310.1	-134258.63	869.07	-760.3223	2.021

TABLE II. - Concluded. SAMPLE OUTPUT

```
                     AFTERMIXING PROPERTIES

            NOZZLE WITH NO BOUNDARY LAYER CORRECTION

  XMFS1 =1.7666  SPACING =.090740  TE =0.       XM2 =1.7532  V/VCR1 = 1.523  XMX1 = 0.604  XMX2 = 0.593

  ALPH1= 70.001  ALPH2= 70.241  PT2/PTO= 0.9861  PTO/P2=    5.407  T2/TTO= 0.6253  V/VCR2= 1.515  EBAR2=0.00651  ETA-V=0.9935
6
            NOZZLE WITH BOUNDARY LAYER CORRECTION

  XMFS1 =1.7666  SPACING =.092147  TE =0.       XM2 =1.7533  V/VCR1 = 1.523  XMX1 = 0.604  XMX2 = 0.593

  ALPH1= 69.999  ALPH2= 70.237  PT2/PTO= 0.9862  PTO/P2=    5.407  T2/TTO= 0.6252  V/VCR2= 1.516  EBAR2=0.00643  ETA-V=0.9936

  DELSR(N+1) =0.000315  DELSR(N+2) =0.000315  THET(N+1) =0.000065  THET(N+2) =0.000065
```

NOZZLE COORDINATES SUPERSONIC SECTION

NOZZLE WITH BOUNDARY LAYER CORRECTION

7

X	Y	XTU	YTU	XTL	YTL
0.	0.011068	-0.010400	0.003786	0.010400	-0.003786
0.012425	0.013261	-0.008211	0.016212	0.016712	0.007140
0.014760	0.013652	-0.007780	0.018539	0.017877	0.009200
0.016521	0.013931	-0.007439	0.020289	0.018741	0.010759
0.018049	0.014159	-0.007131	0.021803	0.019478	0.012118
0.019455	0.014356	-0.006835	0.023192	0.020144	0.013371
0.020789	0.014531	-0.006543	0.024505	0.020765	0.014565
0.022078	0.014689	-0.006251	0.025771	0.021354	0.015723
0.023341	0.014832	-0.005953	0.027007	0.021921	0.016860
0.024590	0.014963	-0.005649	0.028224	0.022471	0.017988
0.025831	0.015082	-0.005337	0.029432	0.023007	0.019114
0.027072	0.015190	-0.005014	0.030635	0.023534	0.020243
0.028317	0.015288	-0.004680	0.031838	0.024051	0.021380
0.029571	0.015375	-0.004333	0.033046	0.024562	0.022529
0.030837	0.015452	-0.003972	0.034262	0.025068	0.023691
0.032117	0.015519	-0.003597	0.035488	0.025568	0.024871
0.033414	0.015575	-0.003207	0.036726	0.026065	0.026071
0.034731	0.015621	-0.002799	0.037979	0.026558	0.027292
0.036068	0.015656	-0.002375	0.039248	0.027049	0.028538
0.037429	0.015679	-0.001931	0.040535	0.027536	0.029809
0.038815	0.015685	-0.001462	0.041839	0.028015	0.031109
0.047342	0.015703	0.001438	0.049858	-0.000000	-0.000000
0.055869	0.015721	0.004337	0.057876	-0.000000	-0.000000
0.064396	0.015738	0.007238	0.065895	-0.000000	-0.000000
0.072922	0.015753	0.010140	0.073912	-0.000000	-0.000000
0.081449	0.015768	0.013043	0.081930	-0.000000	-0.000000
0.089976	0.015782	0.015946	0.089947	-0.000000	-0.000000
0.098503	0.015795	0.018851	0.097964	-0.000000	-0.000000
0.107030	0.015807	0.021756	0.105981	-0.000000	-0.000000
0.115556	0.015819	0.024661	0.113997	-0.000000	-0.000000
0.124083	0.015830	0.027567	0.122013	-0.000000	-0.000000
0.125000	0.015832	0.027879	0.122875	-0.000000	-0.000000
0.125405	0.015832	0.028017	0.123256	-0.000000	-0.000000

PROGRAM DESCRIPTION

The program SSN is divided into two main parts, subroutines NOZZL and BL. This division was necessary because of the storage requirements of the IBM 7094/7044 direct coupled system which has a 32 768 word core (77777_8). An overlay system was used with SSN as the common link between NOZZL and BL. NOZZL requires $51216_{(8)}$ words of storage and BL requires $43535_{(8)}$ words of storage. SSN uses $14134_{(8)}$ words of storage. All quantities calculated in NOZZL and needed by BL are stored in SSN through two common blocks and the CALL statement.

Subroutine NOZZL

This subroutine designs a loss-free supersonic nozzle by the method of characteristics. The description of the method, program, and main dictionary of variables is given in reference 1. The length of the straight section on the suction surface of the nozzle is determined by the nozzle angle.

The calling sequence for NOZZL is as follows:

CALL NOZZL(X, Y, POPTZ, SP, NST, NCP, ALP1, GAM, NSP)

where

X	X-coordinate of nozzle contour point (output)
Y	Y-coordinate of nozzle contour point (output)
POPTZ	static to total pressure ratio (output)
SP	spacing (output)
NST	total number of points (output)
NCP	number of points excluding the ten points for the straight section (output)
ALP1	nozzle flow angle at station 1, measured from tangential direction (output)
GAM	specific heat ratio (output)
NSP	number of subsonic points including throat point (output)

The program variables for NOZZL that are not in reference 1 are

C	chord based on throat half-height of unity
CH	chord, m (ft)
M	Mach number
PSPT (NC)	static to total pressure ratio at a contour point
PTPS (NC)	total to static pressure ratio at a contour point
S	spacing, m (ft)
Z	length of straight section, m (ft)

Subroutine BL

This subroutine calculates the boundary-layer characteristics, mixing losses, and corrected nozzle contour. The major subroutines for BL are INPUT, PRECAL, LAMNAR, TURBLN, PROFIL, AFMIX, and NOZZLC. These, in turn, call several other subroutines. All these subroutines and their relation are shown in figure 3.

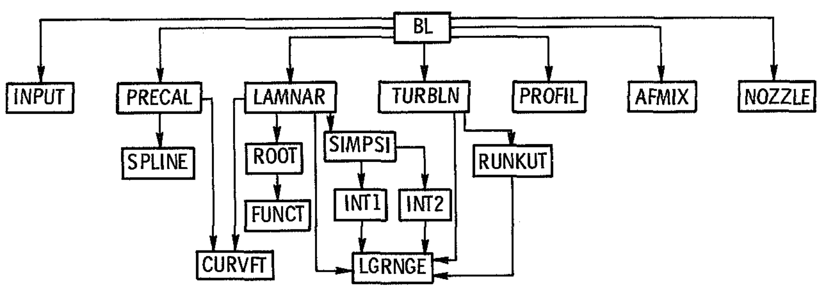

Figure 3. - Calling relation of BL subroutines.

Subroutines INPUT, PRECAL, LAMNAR, TURBLN, and PROFIL. - These subroutines and their auxiliary subroutines shown in figure 3 calculate the boundary-layer characteristics. The methods used for calculating the boundary-layer characteristics and the program description are given in reference 3.

Since a highly favorable pressure gradient exists, the range of the equations of reference 5 was extended by the method given in reference 5. These changes were made in LAMNAR and are for SW (temperature function at the wall) = 0. The changes are noted in the program with comment cards. Two curve fit ranges were also extended. RCRIT and DIFF were extended as follows:

RCRIT = 8.3163, when SHAPK is greater than 0.07

DIFF = 44 000 KBAR + 700, when KBAR is greater than 0.03

Subroutine AFMIX. - Subroutine AFMIX takes the boundary-layer parameters and free-stream conditions at the nozzle exit and calculates the aftermixing conditions by the method given in reference 4. In this loss model, the flow sufficiently downstream of the blade row is assumed to be mixed to a uniform condition. The aftermixing conditions are calculated for two cases - nozzle without boundary-layer correction, and nozzle with boundary-layer correction.

For the case with no boundary-layer correction, the displacement and momentum thicknesses on the suction and pressure surface and the free-stream conditions at nozzle exit (station 1) are used to calculate the aftermixing conditions (station 2).

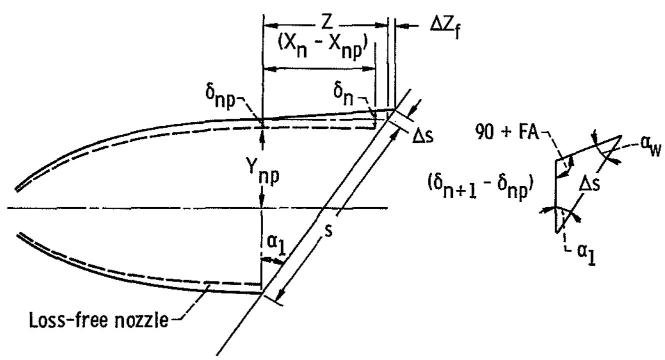

Figure 4. - Quantities required for suction surface calculations at nozzle exit.

For the case with boundary-layer correction, the displacement and momentum thicknesses on the suction surface at the nozzle exit must be calculated. When the displacement thickness is added to the nozzle profile, the length of straight section on the suction surface must be increased to keep the nozzle angle constant. Therefore, from figure 4,

$$Z = 2(Y_{np} + \delta_{np}) \tan \alpha_1 \tag{1}$$

The change in length of straight section is

$$\Delta Z = Z - (X_n - X_{np}) \tag{2}$$

The coordinates of the straight section at this point are

$$X_{n+1} = X_n + \Delta Z \tag{3a}$$

$$Y_{n+1} = Y_{np} \tag{3b}$$

The increase in momentum thickness and displacement thickness is approximately linear with distance on the straight section. Therefore, the displacement angle is

$$FA = \arctan\left(\frac{\delta_n - \delta_{n-2}}{S_n - S_{n-2}}\right) \tag{4}$$

The displacement thickness is

$$\delta_{n+1} = \delta_n + \Delta Z \tan FA \tag{5}$$

Similarly,

$$FA\theta = \arctan\left(\frac{\theta_n - \theta_{n-2}}{S_n - S_{n-2}}\right) \tag{6}$$

Therefore, the momentum thickness is

$$\theta_{n+1} = \theta_n + \Delta Z \tan FA\theta \tag{7}$$

As can be seen from figure 4, the length of the straight section has to be increased further to account for the boundary-layer addition. The angle at the nozzle exit at suction surface is

$$\alpha_w = 180^\circ - \alpha_1 - (90^\circ + FA) \tag{8}$$

From the law of sines

$$\frac{\Delta s}{\sin (90 + FA)} = \frac{\delta_{n+1} - \delta_{np}}{\sin \alpha_w}$$

The change in spacing is

$$\Delta s = (\delta_{n+1} - \delta_{np}) \frac{\sin (90 + FA)}{\sin \alpha_w} \tag{9}$$

The corrected spacing is

$$s_c = s + \Delta s \tag{10}$$

The change in axial length is

$$\Delta Z_f = \frac{\sin \alpha_1}{\sin \alpha_w} (\delta_n - \delta_{np}) \cos FA \tag{11}$$

Therefore, the displacement and momentum thicknesses on the suction surface at the nozzle exit are

$$\delta_{n+2} = \delta_{n+1} + \Delta Z_f \tan FA \tag{12}$$

$$\theta_{n+2} = \theta_{n+1} + \Delta Z_f \tan FA\theta \tag{13}$$

The coordinates of the straight section at this point are

$$X_{n+2} = X_{n+1} + \Delta Z_f \tag{14a}$$

$$Y_{n+2} = Y_{np} \tag{14b}$$

These values of the displacement and momentum thicknesses are used for the suction surface in the calculation of the mixing losses.

The calling sequence for AFMIX is

CALL AFMIX (ALPH1, TE, SP, ME(NST), NP)

where

ALPH1 nozzle flow angle at station 1, measured from axial direction (input)

TE trailing-edge thickness (input)

SP spacing (input)

ME(NST) free-stream Mach number at nozzle exit, station 1

NP	number of points to end of bell portion of the nozzle (last point on pressure surface)

The program variables for AFMIX are

A	$1 - \delta^* - \delta_{te} - \theta^*$
A1	$1 - \delta^* - \delta_{te}$
AFS1	eq. (B4) of ref. 4
ALPH1	see input
ALPW	α_w, fig. 4
C	eq. (C16) of ref. 4
D	eq. (C18) of ref. 4
DELP	displacement thickness on pressure surface
DELS	displacement thickness on suction surface
DELSR	displacement thickness
DELSRT	δ^*, eq. (18a) of ref. 4
DELZ	ΔZ
DELZF	ΔZ_f
DENCH	part of eq. (C21) of ref. 4
DENSR1	$(\rho/\rho')_1$
DENSR2	eq. (C21) of ref. 4
DS	change in spacing, Δs
DF	change in surface length
DTE	δ_{te}, eq. (1) of ref. 4
EBAR	nozzle kinetic energy loss coefficient, eq. (24) of ref. 4
ETAN	nozzle efficiency
FA	displacement thickness angle, eq. (4)
FAMT	momentum thickness angle, eq. (6)
GAM	specific heat ratio
KODE	program control for different solutions
N	NST
NP	see input

PR2 $(p/p')_2$

PT2PTO eq. (C22) of ref. 4

PTOP2 p'_o/p_2

SP spacing, s

THET momentum thickness

THETA θ^*, eq. (18b) of ref. 4

THETS momentum thickness on suction surface

THETP momentum thickness on pressure surface

VXVCR2 ratio of axial velocity to critical velocity at station 2, eq. (C20) of ref. 4

XX $s \cos \alpha_1$

XXX X-coordinate

YYY Y-coordinate

Z Z, length of straight section

Subroutine NOZZLC. - This subroutine calculates the corrected supersonic nozzle coordinates by adding the displacement thickness to the ideal nozzle coordinates. The coordinates are calculated for the nozzle with the nozzle axis horizontal. The coordinates are also calculated for the nozzle with the nozzle axis at the nozzle angle.

The calling sequence for NOZZLC is

CALL NOZZLC (NSP, ALPH1, NP)

where

NSP number of subsonic points including throat point

ALPH1 nozzle angle

NP last point on pressure surface

The program variables are

XSUB length of subsonic portion, m (ft)

XXT X-coordinate of the suction surface (nozzle axis at nozzle angle), m (ft)

XXTL X-coordinate of pressure surface (axis at angle), m (ft)

XXX X-coordinate (nozzle axis horizontal), m (ft)

YYC Y-coordinate of suction surface (axis horizontal), m (ft)

YYCL Y-coordinate of pressure surface (axis horizontal), m (ft)

YYT Y-coordinate suction surface (axis at angle), m (ft)

YYTL Y-coordinate pressure surface (axis at angle), m (ft)

PROGRAM LISTING

```
$IBFTC SSN      DECK

C     SUPERSONIC NOZZLE
       COMMON/C1/GAM,R,PTZ,TTZ,UPMACH,NST,NVP,NTURB,KPVM,KEM,KSMTH,
      1KSPLN,KLE,KATCH,CTHET,DLAM,TLAM,DTURB,TTURB,KPRE,KGRAD,KSDE,KLAM,
      2KMAIN,KPROF,X(100),Y(100),PRES(100),UE(100),ME(100),POPTZ(100),
      3VOVCR(100),TWAL(100)
       COMMON/C2/PSZ,TSZ,UZ,ASZ,ATZ,RHSZ,RHTZ,MUSZ,MUTZ,NUSZ,NUTZ,CP,
      1PR,TC,ARCL
       REAL ME
1      CALL NOZZL(X,Y,POPTZ,SP,NST,NCP,ALP1,GAM,NSP)
       CALL BL(SP,NCP,ALP1,NSP)
       GO TO 1
       END
```

```
$IBFTC NOZZLE  DECK

       SUBROUTINE NOZZL(XCP,YCP,PSPT,S   ,NOP,NCP,ALP1,GAM,NS)
C      TWO DIMENSIONAL   SUPERSONIC NOZZLE WITH A SHARP-EDGED THROAT
       DIMENSION XC(1500),YC(1500),PHI(1500),U(3000),M(3000),
      1X(1500,2),Y(1500,2), PTPS(3000),XCP(100),YCP(100),PSPT(100)
       REAL M,MC,ME,MEC,MT
4      READ(5,3) ME,DV,GAM,NP,NS,ALP1,YTH
3      FORMAT(3F6.3,2I2,F6.3,F10.5)
       READ(5,9)(XCP(I),YCP(I),PSPT(I),I=1,NS)
9      FORMAT(3F10.1)
       NCP=NS+1
       ALP1=ALP1*.017453
       VE=SQRT((GAM+1.0)/(GAM-1.0))*ATAN(SQRT((GAM-1.0)/(GAM+1.0)
      1*(ME**2-1.0))) - ATAN(SQRT(ME**2-1.0))
       DV= DV*.017453
       XT=0.0
       YT=1.0
       MT =1.0
       PTPST = ((GAM+1.0)/2.0)**(GAM/(GAM-1.0))
       KMAX = INT(.5*VE/DV+1.5)
       DV=VE/(2.0*FLOAT(KMAX-1))
       NMAX = KMAX
       WRITE(6,251)
```

```
251   FORMAT(1H1,36X,59HTWO DIMENSIONAL SUPERSONIC NOZZLE WITH A SHARP-E
     1DGED THROAT)
      VE= VE*57.2958
      DV =DV*57.2958
      ALP1 = ALP1*57.2958
      WRITE(6,252) VE,ME  ,GAM,DV,YTH,ALP1
252   FORMAT( 1H0,4HVE =,F8.3,10X,4HME =,F8.3,10X,7HGAMMA =,F6.3,10X,
     19HDELTA V =,F7.5, 8X,5HYTH =,E12.5,8X,6HALP1 =,F7.3)
      WRITE(6,247)
247   FORMAT( 1H0,5X,24HSUBSONIC SECTION (INPUT))
      WRITE(6,248)
248   FORMAT( 1H0,5X,1HX,14X,1HY,14X,4HPSPT)
      WRITE( 6,255) (XCP(I),YCP(I),PSPT(I),I=1,NS)
255   FORMAT( 1H ,F10.5,2F15.5)
      WRITE(6,246)
246   FORMAT( 1H0,5X,18HSUPERSONIC SECTION)
      WRITE(6,253)
253   FORMAT( 1H0,5X,1HX,14X,1HY,14X,1HM,14X,4HPTPS)
      WRITE( 6,254)XT,YT,MT,PTPST
      ALP1 = ALP1* .017453
      DV= DV*.017453
C     FLOW ANGLE CALCULATION
      DO 5 K=1,KMAX
5     PHI(K) = FLOAT(K-1)*DV
C     MACH ANGLE CALCULATION
      NK = 2*KMAX-1
      DO 6 I=1,NK
6     CALL UA(I,DV,GAM,U(I))
C     REGION NEAR THE THROAT
      DO 100 K=1,KMAX
      NC = 1
      N=1
      I = K+2*(NC-1)
      IF (K.NE.1) GO TO 10
      SLOPE1 = -TAN((U(I)+U(I+1))/2.0-(PHI(K)+PHI(K+1))/2.0)
      X(K,N)= - 1.0/SLOPE1
      Y(K,N) = 0.0
      GO TO 100
10    IF (K.EQ.KMAX) GO TO 20
      SLOPE1 = -TAN((U(I)+U(I+1))/2.0-(PHI(K)+PHI(K+1))/2.0)
      SLOPE2 =  TAN((U(I)+U(I+1))/2.0+(PHI(K)+PHI(K-1))/2.0)
      X(K,N) = (1.0-(Y(K-1,N)-SLOPE2*X(K-1,N)))/
     1(SLOPE2-SLOPE1)
      Y(K,N)=Y(K-1,N)+SLOPE2*(X(K,N)-X(K-1,N))
      GO TO 100
20    SLOPE1= TAN(PHI(K))
      SLOPE2 =  TAN((U(I)+U(I+1))/2.0+(PHI(K)+PHI(K-1))/2.0)
      X(K,N) = (1.0-(Y(K-1,N)-SLOPE2*X(K-1,N)))/
     1(SLOPE2-SLOPE1)
      Y(K,N)=Y(K-1,N)+SLOPE2*(X(K,N)-X(K-1,N))
      XC(NC)=X(KMAX,N)
      YC(NC)= Y(KMAX,N)
      M(NC) = 1.0/SIN(U(I))
      PTPS(NC) =(1.0+(GAM-1.0)*M(NC)**2/2.0)**(GAM/(GAM-1.0))
100   CONTINUE
      IF (MOD(NC,NP) .EQ.0) WRITE(6,254) XC(NC),YC(NC),M(NC),PTPS(NC)
      IF (MOD(NC,NP).NE.0) GO TO 203
      XCP(NCP)=XC(NC)
```

```
      YCP(NCP)=YC(NC)
      PSPT(NCP) = 1.0/PTPS(NC)
      NCP=NCP+1
C     REGION DOWNSTREAM OF THE THROAT
203    CONTINUE
205   N=2
      KMAX=KMAX-1
      NC=NC+1
      DO 200  K=1,KMAX
      I = K+2*(NC-1)
      IF (K.NE.1) GO TO 201
      SLOPE1 = -TAN((U(I)+U(I+1))/2.0-(PHI(K)+PHI(K+1))/2.0)
      X(K,N)=-(Y(K+1,N-1)-SLOPE1*X(K+1,N-1))/SLOPE1
      Y(K,N)=0.0
      GO TO 200
201   IF(K.EQ.KMAX) GO TO 202
      SLOPE1 = -TAN((U(I)+U(I+1))/2.0-(PHI(K)+PHI(K+1))/2.0)
      SLOPE2 =  TAN((U(I)+U(I+1))/2.0+(PHI(K)+PHI(K-1))/2.0)
      X(K,N)=((Y(K+1,N-1)-SLOPE1*X(K+1,N-1))-(Y(K-1,N)-SLOPE2
     1*X(K-1,N)))/(SLOPE2 - SLOPE1)
      Y(K,N)=Y(K-1,N)+SLOPE2*(X(K,N)-X(K-1,N))
      GO TO 200
202   SLOPE1=TAN(PHI(K))
      SLOPE2 =  TAN((U(I)+U(I+1))/2.0+(PHI(K)+PHI(K-1))/2.0)
      X(K,N)=((Y(K+1,N-1)-SLOPE1*X(K+1,N-1))-(Y(K-1,N)-SLOPE2
     1*X(K-1,N)))/(SLOPE2 - SLOPE1)
      Y(K,N)=Y(K-1,N)+SLOPE2*(X(K,N)-X(K-1,N))
      XC(NC)=X(KMAX,N)
      YC(NC)=Y(KMAX,N)
      M(NC) = 1.0/SIN(U(I))
      PTPS(NC) =(1.0+(GAM-1.0)*M(NC)**2/2.0)**(GAM/(GAM-1.0))
200   CONTINUE
      IF (MOD(NC,NP) .EQ.0) WRITE(6,254) XC(NC),YC(NC),M(NC),PTPS(NC)
      IF (MOD(NC,NP).NE.0) GO TO 249
      XCP(NCP)=XC(NC)
      YCP(NCP)=YC(NC)
      PSPT(NCP) = 1.0/PTPS(NC)
      NCP=NCP+1
249   IF (KMAX.EQ.2) GO TO 250
      DO 204 K=1,KMAX
      X(K,1)=X(K,2)
204   Y(K,1)=Y(K,2)
      GO TO 205
250   NMA=NMAX-1
      IF(XCP(NCP-1).EQ.XC(NMA)) GO TO 300
      XCP(NCP)=XC(NMA)
      YCP(NCP)=YC(NMA)
       PSPT(NCP)=1.0/PTPS(NMA)
      GO TO 301
300   NCP=NCP-1
301   WRITE (6,254) XC(NMA),YC(NMA),M(NMA),PTPS(NMA)
254   FORMAT(1H ,F10.5,F15.5,F15.5,F15.7)
      Z=2.0*YC(NMA)/TAN(ALP1)*YTH
      S = 2.0*YC(NMA)/SIN(ALP1)*YTH
      C=XCP(NS)+XC(NMA)+Z/YTH
      CH = C*YTH
      PSPT(NS) = 1.0/PTPST
      NOP=NCP+10
      DO 304 I=1,NOP
```

```
      IF (I.GE.(NS+1)) GO TO 302
      XCP(I)=XCP(I)/C*CH
      YCP(I)=YCP(I)/C*CH
      GO TO 304
302   IF(I.GE.(NCP+1)) GO TO 303
      XCP(I)=XCP(I)/C*CH+XCP(NS)
      YCP(I)=YCP(I)/C*CH
      GO TO 304
303   PSPT(I)= PSPT(NCP)
      XCP(I) =Z/10.0+XCP(I-1)
      YCP(I)=YCP(NCP)
304   CONTINUE
      RETURN
      END
```

```
$IBFTC U        DECK

      SUBROUTINE UA(I,DV,GAM,U)
      REAL MC,M,MC1
      X2M=SQRT((GAM+1.0)/(GAM-1.0))-.01
      V = FLOAT(I-1)*DV
      IF(V.EQ.0.0) GO TO 5
      EXTERNAL  VA
      IF (MC.NE.1.0) GO TO 7
      DVDMC=.02
      GO TO 4
7     A= SQRT((GAM+1.0)/(GAM-1.0))
      B= (GAM-1.0)*MC**2-GAM
      C= (GAM+1.0)/MC**2-GAM
      DVDMC= A*(GAM-1.0)*MC/(1.0-B**2)**(.5)-MC**(-3)*(GAM+1.0)/(1.0-
     1C**2)**(.5)
4     MC= (V-VA(MC))/DVDMC+MC
      IF (MC.LT.X2M) GO TO 9
      MC=X2M
      WRITE (6,2) MC,NC,K
2     FORMAT(1H0,41HLIMIT HAS BEEN REACHED.    MC SET = TO X2M,5X,4HMC =,
     1F8.3,5X,4HNC =,I4,5X,3HK =,I4)
9     V1= VA(MC)
      IF (ABS(V-V1).LT. .00001) GO TO 8
      GO TO 7
8     M = SQRT(((2.0/(GAM+1.0))*MC**2)/(1.0-((GAM-1.0)/
     1(GAM+1.0)*MC**2)))
      GO TO 6
5     M=1.0
      MC=1.0
6     U=ARSIN(1.0/M)
      RETURN
      END
```

$IBFTC VAN   DECK

```
      FUNCTION VA(X)
      COMMON/C1/GAM,R,PTZ,TTZ,UPMACH,NST,NVP,NTURB,KPVM,KEM,KSMTH,
     1KSPLN,KLE,KATCH,CTHET,DLAM,TLAM,DTURB,TTURB,KPRE,KGRAD,KSDE,KLAM,
     2KMAIN,KPROF,X(100),Y(100),PRES(100),UE(100),ME(100),POPTZ(100),
     3VOVCR(100),TWAL(100)
      A= SQRT((GAM+1.0)/(GAM-1.0))-1.0
      B= ARSIN((GAM-1.0)*X**2-GAM)
      C=ARSIN((GAM+1.0)/X**2-GAM)
      VA=(3.1415926)/4.0 * A+.5*((A+1.0)*B+C)
      RETURN
      END
```

$IBFTC BLAYER DECK

```
      SUBROUTINE BL(SP,NP,ALP1,NSP)
      COMMON/C1/GAM,R,PTZ,TTZ,UPMACH,NST,NVP,NTURB,KPVM,KEM,KSMTH,
     1KSPLN,KLE,KATCH,CTHET,DLAM,TLAM,DTURB,TTURB,KPRE,KGRAD,KSDE,KLAM,
     2KMAIN,KPROF,X(100),Y(100),PRES(100),UE(100),ME(100),POPTZ(100),
     3VOVCR(100),TWAL(100)
      COMMON/C2/PSZ,TSZ,UZ,ASZ,ATZ,RHSZ,RHTZ,MUSZ,MUTZ,NUSZ,NUTZ,CP,
     1PR,TC,ARCL
      COMMON/C3/XOM(100),YOM(100),S(100),SOL(100),AE(100),TSE(100),
     1TAWL(100),TAWT(100),TBAR(100),RW(100),SW(100),SUTHL(100),
     2RHSW(100),RHSE(100),HEADW(100),HEADE(100),NUW(100),MUBAR(100),
     3AA(100),BB(100),FF(100),DUDS(100),DMDS(100),DMDL(100)
      COMMON/C4/THET(100),DELSR(100),DELTA(100),FORM(100),
     1FORMI(100),FORMTR(100),RTH(100),RTHI(100),CF(100),
     1TAUW(100),NUSS(100),DTDY(100),HTRAN(100),CRN(100)
      COMMON/C5/ERROR,TRANS,SEPRN
      REAL ME
      LOGICAL ERROR,TRANS,SEPRN
   10 CALL INPUT
      READ (5,99) TE
   99 FORMAT(F6.1)
      IF (ERROR) GO TO 10
      CALL PRECAL
      IF(ERROR) GO TO 10
      CALL LAMNAR
      IF (ERROR) GO TO 10
      IF (SEPRN) GO TO 20
      IF (.NOT.TRANS) GO TO 20
      CALL TURBLN
      IF (ERROR) GO TO 10
   20 CALL PROFIL
      ALPH1 = 3.14159/2.0 - ALP1
      CALL AFMIX(ALPH1,TE,SP,ME(NST),NP)
      CALL NOZZLC(NSP,ALPH1,NP)
      GO TO 10
      END
```

```
$IBFTC INPU    DECK
      SUBROUTINE INPUT
      COMMON/C1/GAM,R,PTZ,TTZ,UPMACH,NST,NVP,NTURB,KPVM,KEM,KSMTH,
     1KSPLN,KLE,KATCH,CTHET,DLAM,TLAM,DTURB,TTURB,KPRE,KGRAD,KSDE,KLAM,
     2KMAIN,KPROF,X(100),Y(100),PRES(100),UE(100),ME(100),POPTZ(100),
     3VOVCR(100),TWAL(100)
      COMMON/C5/ERROR,TRANS,SEPRN
      LOGICAL ERROR,TRANS,SEPRN
      REAL ME
      ERROR= .FALSE.
      TRANS= .FALSE.
      SEPRN= .FALSE.
      CTHET = 0.0
      KATCH = 0
      KSPLN = 1
      KLE = 0
      WRITE(6,1000)
      READ (5,1020)      R,PTZ,TTZ,UPMACH
      WRITE(6,1050)
      WRITE(6,1060)      R,PTZ,TTZ,UPMACH
      READ(5,1010)NVP,NTURB,KEM
      WRITE(6,1070)NST,NVP,NTURB,KEM
      READ (5,1020) DLAM,TLAM,DTURB,TTURB
      WRITE(6,1080) DLAM,TLAM,DTURB,TTURB
      READ (5,1010) KPRE,KGRAD,KSDE,KLAM,KMAIN,KPROF
      WRITE(6,1090) KPRE,KGRAD,KSDE,KLAM,KMAIN,KPROF
      IF(NST.GT.100.OR.NTURB.GT.NST.OR.KEM.LT.0.OR.KEM.GT.1.OR.KSPLN.LT.
     10.OR.KSPLN.GT.1.OR.KLE.LT.0.OR.KLE.GT.1.OR.KATCH.LT.0.OR.KATCH.GT.
     21) GO TO 70
      DO 1 I =1,NST
1     TWAL(I) = TTZ
      WRITE(6,1130) (X(I),Y(I),POPTZ(I),TWAL(I),I=1,NST)
      RETURN
   70 ERROR = .TRUE.
      WRITE(6,1170)
      RETURN
 1000 FORMAT(1H1////)
 1010 FORMAT(16I5)
 1020 FORMAT(8F10.5)
 1030 FORMAT(4F10.5)
1050  FORMAT(1H0, 6X,22HBOUNDARY LAYER SECTION)
 1060 FORMAT(//              9X,1HR,11X,3HPTZ,8X,3HTTZ,8X,6HUPMACH/        3X
     1,F9.2,3X,F10.2,2X,F9.2,4X,F8.4)
 1070 FORMAT(/6X,3HNST,8X,3HNVP,9X,5HNTURB,7X,3HKEM/6X,I3,8X,I3,10X,
     1 I3,9X,I2)
 1080 FORMAT(/6X,4HDLAM,7X,4HTLAM,8X,5HDTURB,7X,5HTTURB/4X,F10.6,1X,F10.
     16,2X,F10.6,2X,F10.6)
 1090 FORMAT(/6X,4HKPRE,7X,5HKGRAD,7X,4HKSDE,8X,4HKLAM,7X,5HKMAIN,7X,5HK
     1PROF/7X,I2,9X,I2,10X,I2,10X,I2,9X,I2,10X,I2)
 1130 FORMAT(/9X,1HX,11X,1HY,10X,5HPOPTZ,10X,4HTWAL/(3X,F10.5,2X,F10.5,4
     1X,F10.6,4X,F10.4))
 1170 FORMAT(////10X,48HERROR IN INPUT DATA.  RECHECK INPUT INSTRUCTIONS
     1)
      END
```

```
$IBFTC PRECA   DECK
      SUBROUTINE PRECAL
      COMMON/C1/GAM,R,PTZ,TTZ,UPMACH,NST,NVP,NTURB,KPVM,KEM,KSMTH,
     1KSPLN,KLE,KATCH,CTHET,DLAM,TLAM,DTURB,TTURB,KPRE,KGRAD,KSDE,KLAM,
     2KMAIN,KPROF,X(100),Y(100),PRES(100),UE(100),ME(100),POPTZ(100),
     3VOVCR(100),TWAL(100)
      COMMON/C2/PSZ,TSZ,UZ,ASZ,ATZ,RHSZ,RHTZ,MUSZ,MUTZ,NUSZ,NUTZ,CP,
     1PR,TC,ARCL
      COMMON/C3/XOM(100),YOM(100),S(100),SOL(100),AE(100),TSE(100),
     1TAWL(100),TAWT(100),TBAR(100),RW(100),SW(100),SUTHL(100),
     2RHSW(100),RHSE(100),HEADW(100),HEADE(100),NUW(100),MUBAR(100),
     3AA(100),BB(100),FF(100),DUDS(100),DMDS(100),DMDL(100)
      COMMON/C9/ERROR,TRANS,SEPRN
      DIMENSION SDER(100),CMU(20),CPR(20),CTC(20)
      REAL MUSZ,MUTZ,NUSZ,NUTZ,MUSLE,MUSLM,ME,NUW,MUBAR
      LOGICAL ERROR,TRANS,SEPRN
C
C  READ DATA FOR MU, PR, AND TC CURVE FITS
C
      DATA(CMU(I),I=1,5)/-.01945170,1.3019531,-.34511323,
     1.068277826,-.00566593/
      DATA(CPR(I),I=1,5)/.8557,-.234136,.1078624,
     1-.0236214,.00202863/
      DATA(CTC(I),I=1,5)/-.03839323,1.2697427,-.30911252,
     1.08743781,-.009674725/
C
C  INITIALIZE STATIC AND TOTAL PARAMETERS
C
      TSLE= 518.688
      TSLM= 288.160
      MUSLE= 3.711402E-7
      MUSLM= 1.777029E-5
      TCSLE= 3.202206E-3
      TCSLM= 2.561796E-2
      TSZ= TTZ/(1.+(GAM-1.)/2.*UPMACH**2)
      PSZ= PTZ*(TSZ/TTZ)**(GAM/(GAM-1.))
      RHSZ= PSZ/R/TSZ
      RHTZ= PTZ/R/TTZ
      ASZ= SQRT(GAM*R*TSZ)
      ATZ= SQRT(GAM*R*TTZ)
      UZ= UPMACH*ASZ
      CP= R*GAM/(GAM-1.)
      IF (KEM.EQ.1) GO TO 10
      TCON= 198.60
      TR1= TSZ/TSLE
      TR2= TTZ/TSLE
      GO TO 20
   10 TCON= 110.33
      TR1= TSZ/TSLM
      TR2= TTZ/TSLM
   20 CALL CURVFT(CPR,PR,TR1,0,4,0)
      CALL CURVFT(CTC,TC,TR1,0,4,0)
      CALL CURVFT(CMU,MUSZ,TR1,0,4,0)
      CALL CURVFT(CMU,MUTZ,TR2,0,4,0)
      IF (KEM.EQ.1) GO TO 30
```

```
      TC= TC*TCSLE
      MUSZ= MUSZ*MUSLE
      MUTZ= MUTZ*MUSLE
      GO TO 40
   30 TC= TC*TCSLM
      MUSZ= MUSZ*MUSLM
      MUTZ= MUTZ*MUSLM
   40 NUSZ= MUSZ/RHSZ
      NUTZ= MUTZ/RHTZ
C
C  CALCULATE GEOMETRY RATIOS AND ARC LENGTHS
C
      XOM(1)= X(1)/X(NST)
      YOM(1)= Y(1)/X(NST)
      S(1)= 0.
      DO 50 I=2,NST
      XOM(I)= X(I)/X(NST)
      YOM(I)= Y(I)/X(NST)
   50 S(I)= S(I-1)+SQRT((X(I)-X(I-1))**2+(Y(I)-Y(I-1))**2)
      ARCL= S(NST)
      DO 60 I=1,NST
   60 SOL(I)= S(I)/ARCL
C
C  CALCULATE PRES,UE,ME,POPTZ,AND VOVCR AT EACH STATION
C
C  PRESSURE OVER TOTAL PRESSURE GIVEN AS INPUT
  130 DO 140 I=1,NST
      IF(POPTZ(I).LT.0..OR.POPTZ(I).GT.1.)GO TO 290
      PRES(I)= POPTZ(I)*PTZ
      UE(I)= SQRT(2.*GAM/(GAM-1.)*PTZ/RHTZ*(1.-(PRES(I)/PTZ)**((GAM-1.)/
     1GAM)))
      TSE(I)= TTZ-UE(I)**2/(2.*CP)
      AE(I)= SQRT(GAM*R*TSE(I))
      ME(I)= UE(I)/AE(I)
  140 VOVCR(I)= SQRT((GAM+1.)/(GAM-1.)*(1.-(PRES(I)/PTZ)**((GAM-1.)/GAM)
     *))
C
C  PRINT INITIAL CALCULATED PARAMETERS
C
  170 WRITE(6,1000)
      WRITE(6,1010) PSZ,TSZ,UZ,ASZ,ATZ,RHSZ,RHTZ,MUSZ,MUTZ,NUSZ,NUTZ,CP,
     1PR,TC,ARCL
      WRITE(6,1020) (I,PRES(I),UE(I),ME(I),POPTZ(I),VOVCR(I),I=1,NST)
C
C
C  PRINT GEOMETRY PARAMETERS
C
  200 IF (KPRE.NE.1) GO TO 210
      WRITE(6,1030) (I,X(I),Y(I),S(I),XOM(I),YOM(I),SOL(I),I=1,NST)
C
C  CALCULATE OTHER NECESSARY PARAMETERS AT EACH STATION
C
  210 DO 220 I=1,NST
      TEM1= 1.+.5*(GAM-1.)*ME(I)**2
      RHSW(I)= PRES(I)/R/TWAL(I)
      RHSE(I)= PRES(I)/R/TSE(I)
      HEADW(I)= .5*RHSW(I)*UE(I)**2
      HEADE(I)= .5*RHSE(I)*UE(I)**2
      SW(I)= TWAL(I)/TTZ-1.
```

```
      SUTHL(I)= SQRT(TWAL(I)/TTZ)*(TTZ+TCON)/(TWAL(I)+TCON)
      NUW(I)= SUTHL(I)*NUTZ*(1.+SW(I))**2*TEM1**(GAM/(GAM-1.))
      RW(I)= UE(I)*S(I)/NUW(I)
      TAWL(I)= TSE(I)*(1.+PR**(1./2.)*(TEM1-1.))
      TAWT(I)= TSE(I)*(1.+PR**(1./3.)*(TEM1-1.))
      TBAR(I)= .5*(TWAL(I)+TSE(I))+.22*PR**(1./3.)*(TTZ-TSE(I))
      MUBAR(I)= MUTZ*SUTHL(I)*TBAR(I)/TTZ
      BB(I)= ME(I)*ATZ/NUTZ*(TSE(I)/TTZ)**((GAM+1.)/(2.*GAM-2.))
      AA(I)= BB(I)*TSE(I)/TBAR(I)*(MUBAR(I)/MUTZ)**.268
      FF(I)=1.+.1599*ME(I)**2+.60*SW(I)+.2101*SW(I)*ME(I)**2+.0114*ME(I)
     1**4+.018C*SW(I)*ME(I)**4+.1825*SW(I)**2+.0735*SW(I)**2*ME(I)**2
     2+.0073*SW(I)**2*ME(I)**4
  220 CONTINUE
C
C  COMPUTE VELOCITY AND MACH NUMBER GRADIENTS ALONG THE SURFACE
C
      DUDS(NST) = 0.0
      DMDS(NST) = 0.0
      I = 1
  221 IF (I.EQ.NST) GO TO 240
      IF(ME(I) .NE. ME(I+1)) GO TO 222
      DUDS(I) = 0.0
      DMDS(I) = 0.0
      I = I+1
      GO TO 221
  222 I1= I
  223 I = I+1
      IF(I.EQ.NST) GO TO 224
      IF (ME(I).NE.ME(I+1)) GO TO 223
  224 NSP = I+1 -I1
C  SPLINE CURVE TECHNIQUE
  230 CALL SPLINE(S(I1),UE(I1),NSP,DUDS(I1),SDER(I1))
      CALL SPLINE(S(I1),ME(I1),NSP,DMDS(I1),SDER(I1))
      IF (I.NE.NST) GO TO 221
  240 DO 250 I=1,NST
  250 DMDL(I)= ARCL*DMDS(I)
C
C  PRINT OTHER CALCULATED PARAMETERS
C
      IF(KPRE.NE.1) GO TO 260
      WRITE(6,1050) (I,AE(I),TSE(I),TWAL(I),TAWL(I),TAWT(I),TBAR(I),
     II=1,NST)
      WRITE(6,1060) (I,RW(I),SW(I),SUTHL(I),RHSW(I),RHSE(I),HEADW(I),
     IHEADE(I),NUW(I),MUBAR(I),I=1,NST)
  260 IF(KGRAD.NE.1) GO TO 270
      WRITE(6,1070)
      WRITE(6,1080) (I,DUDS(I),DMDS(I),DMDL(I),I=1,NST)
C
C  CHECK FOR IMPROPER INPUT
C
  270 DO 280 I=2,NST
      IF (UE(I).NE.0.) GO TO 280
      ERROR= .TRUE.
      WRITE(6,1090)
      RETURN
  280 CONTINUE
      RETURN
  290 ERROR= .TRUE.
      WRITE(6,1100)
      RETURN
```

```
C
C  FORMAT STATEMENTS
C
 1000 FORMAT(1H1///4X,24HPRELIMINARY CALCULATIONS///)
 1010 FORMAT(5X,10HPSZ      =  F12.5/5X,10HTSZ      =  F10.4/5X,10HUZ      =
     1 F11.5//5X,10HASZ      =  F11.4/5X,10HATZ      =  F11.4//5X,10HRHSZ
     2=  G15.7/5X,10HRHTZ     =  G15.7//5X,10HMUSZ     =  G15.7/5X,10HMUTZ
     3 =  G15.7//5X,10HNUSZ     =  G15.7/5X,10HNUTZ     =  G15.7//5X,10HCP
     4   =  F11.4/5X,10HPR       =  F9.5/5X,10HTC       =  G15.7/5X,10HARCL
     5  =  F8.4///)
 1020 FORMAT(/1X,7HSTATION,7X,4HPRES,13X,2HUE,12X,2HME,11X,5HPOPTZ,9X,5H
     1VOVCR/(2X,I3,5X,F12.5,3X,F12.5,4X,F10.6,4X,F10.6,4X,F10.6))
 1030 FORMAT(///1X,7HSTATION,7X,1HX,12X,1HY,12X,1HS,12X,3HXOM,9X,3HYOM,
     19X,3HSOL/(2X,I3,3X,F12.5,1X,F12.5,1X,F12.5,4X,F9.5,3X,F9.5,3X,F9.5
     2))
 1050 FORMAT(///1X,7HSTATION,5X,2HAE,10X,3HTSE,9X,4HTWAL,8X,4HTAWL,8X,4H
     1TAWT,8X,4HTBAR/(2X,I3,4X,F9.3,5(4X,F8.3)))
 1060 FORMAT(///1X,7HSTATION,11X,2HRW,6X,2HSW,4X,5HSUTHL,7X,4HRHSW,12X,4
     1HRHSE,8X,5HHEADW,4X,5HHEADE,9X,3HNUW,12X,5HMUBAR/(2X,I3,3X,F16.1,2
     2X,F4.1,1X,F7.3,2X,G14.6,2X,G14.6,1X,F8.3,1X,F8.3,2X,G14.6,2X,G14.6
     3))
 1070 FORMAT(1H1///21X,17HSURFACE GRADIENTS///)
 1080 FORMAT(1X,7HSTATION,13X,4HDUDS,15X,4HDMDS,15X,4HDMDL/(2X,I3,4X,F18
     1.6,1X,F18.6,1X,F18.6))
 1090 FORMAT(/////10X,83HTHERE IS A STAGNATION POINT AT A STATION OTHER
     1THAN STATION 1.  THIS IS NOT ALLOWED)
 1100 FORMAT(/////10X,111HAN INPUT PRESSURE,VELOCITY, OR MACH NUMBER IS
     1EITHER LESS THAN ZERO OR GREATER THAN ITS MAXIMUM ALLOWABLE VALUE)
      END
```

```
$IBFTC LAMNA    DECK

      SUBROUTINE LAMNAR
      COMMON/C1/GAM,R,PTZ,TTZ,UPMACH,NST,NVP,NTURB,KPVM,KEM,KSMTH,
     1KSPLN,KLE,KATCH,CTHET,DLAM,TLAM,DTURB,TTURB,KPRE,KGRAD,KSDE,KLAM,
     2KMAIN,KPROF,X(100),Y(100),PRES(100),UE(100),ME(100),POPTZ(100),
     3VOVCR(100),TWAL(100)
      COMMON/C2/PSZ,TSZ,UZ,ASZ,ATZ,RHSZ,RHTZ,MUSZ,MUTZ,NUSZ,NUTZ,CP,
     1PR,TC,ARCL
      COMMON/C3/XOM(100),YOM(100),S(100),SOL(100),AE(100),TSE(100),
     1TAWL(100),TAWT(100),TBAR(100),RW(100),SW(100),SUTHL(100),
     2RHSW(100),RHSE(100),HEADW(100),HEADE(100),NUW(100),MUBAR(100),
     3AA(100),BB(100),FF(100),DUDS(100),DMDS(100),DMDL(100)
      COMMON/C4/THET(100),DELSR(100),DELTA(100),FORM(100),
     1FORMI(100),FORMTR(100),RTH(100),RTHI(100),CF(100),
     1TAUW(100),NUSS(100),DTDY(100),HTRAN(100),CRN(100)
      COMMON/C5/SHAPL(100),SHAPK(100),B,NS
      COMMON/C6/FTRAN,FORMS
      COMMON/C7/INST,ITRAN,ISEP
      COMMON/C9/ERROR,TRANS,SEPRN
      DIMENSION CORLN(100),CORML(100),SHEAR(100),DTH(100)
      DIMENSION CCN(20),CRCR(20),CDIF(20),CSHR(20),CCRN(20),CDTH(20)
      DIMENSION STAB(505),CTAB1(505),CTAB2(505)
      REAL MUSZ,NUSZ,MUTZ,NUTZ,ME,NUW,MUBAR,NUSS,NURW,KBAR,INT1,INT2
      LOGICAL ERROR,TRANS,SEPRN
      EXTERNAL FUNCT,INT1,INT2
```

```
C
C  READ DATA FOR CORLN(1), RCRIT  DIFF, SHEAR, CRN, AND DTH CURVE FITS
C
      DATA(CCN(I),I=1,6)/-.08178,.06670,-.03143,
     1.00873,.01657,-.01052/
      DATA(CRCR(I),I=1,6)/5.47073,43.6053,227.198,
     1-2067.04,-27172.7,13691.2/
      DATA(CDIF(I),I=1,6)/903.785,26365.0,3.85695E+5,
     11.11044E+6,-4.53853E+7,-7.70276E+7/
      DATA(CSHR(I),I=1,16)/.224488,-1.91539,-9.894,-68.13488,
     1-.001512,-1.4768,-10.52925,-152.2781,-.002406,-.015629,
     1-1.45743,-126.23395,.000752,.005385,.917838,-39.40644/
      DATA(CCRN(I),I=1,16)/2.02056,-19.7211,-24.0495,-1400.002,
     1-.050979,-10.88012,62.4419,-5081.76,-.014343,2.279845,
     1129.7008,-6257.848,.0270567,-1.677051,57.4397,-2552.266/
      DATA(CDTH(I),I=1,16)/8.02829,-4.30978,88.8244,36.4336,
     12.71101,-7.42259,242.293,-16.293, -.16394,-7.61942,286.9795,
     164.11186,-.16758,-3.70289,130.8107,111.3276/
C
C  INITIALIZE PARAMETERS
C
      INST = 0
      ITRAN = 0
      ISEP = 0
      CF(1)= 0.
      TAUW(1)= 0.
      NUSS(1)= 0.
      DTDY(1)= 0.
      HTRAN(1)= 0.
      CRN(1)= 0.
      RTRAN= 0.
      KLE = 0
C
C  CHECK CONSISTENCY OF INITIAL VALUES
C
      IF (DLAM.GE.0..AND.TLAM.GE.0..AND.DTURB.GE.0..AND.TTURB.GE.0.)
     1GO TO 10
      ERROR = .TRUE.
      WRITE(6,1000)
      RETURN
   10 IF (NTURB.NE.1) GO TO 30
      ITRAN = 1
      IF (DTURB.GT.0..AND.TTURB.GT.0.) GO TO 20
      ERROR = .TRUE.
      WRITE(6,1010)
      RETURN
   20 IF (UE(1).GT.0.) GO TO 240
      ERROR = .TRUE.
      WRITE(6,1020)
      RETURN
C
C  BEGIN CALCULATION IN LAMINAR REGION - CHECK FOR INITIAL VALUES
C  CALCULATE INITIAL CORRELATION NUMBER
C
   30 IF (DLAM.EQ.0..AND.TLAM.EQ.0.) GO TO 70
      IF (UE(1).GT.0.) GO TO 40
      ERROR = .TRUE.
      WRITE(6,1030)
      RETURN
   40 IF (TLAM.EQ.0.) GO TO 50
```

```
C   INITIAL MOMENTUM THICKNESS WAS GIVEN
       TEM1= 1.+.5*(GAM-1.)*ME(1)**2
       CORML(1)= -ATZ*TLAM**2/NUTZ/SUTHL(1)/ARCL/TEM1**((3.-GAM)/
      1(2.*GAM-2.))
       CORLN(1) = CORML(1)*DMDL(1)
       GO TO 90
C   INITIAL DISPLACEMENT THICKNESS WAS GIVEN
    50 IF (ABS(DMDL(1)).GE..0001) GO TO 60
       CORLN(1)= 0.
       TEM1= 1.+.5*(GAM-1.)*ME(1)**2
       FORM(1)= 2.38411*(1.+(2.79-1.78*PR**.5)*((1.+SW(1))*TEM1-1.))+(4.6
      15*PR**(1./3.)-3.65*PR*.5)*PR**.5*(TEM1-1.)
       THET(1)= DLAM/FORM(1)
       CORML(1)= -ATZ*THET(1)**2/NUTZ/SUTHL(1)/ARCL/TEM1**((3.-GAM)/(2.*G
      1AM-2.))
       GO TO 90
    60 IF (DMDL(1).GT.0.) CALL ROOT(-1.,0.,DLAM,FUNCT,.5E-5,CORLN(1),SL)
       IF (DMDL(1).LT.0.) CALL ROOT( 0.,.2,DLAM,FUNCT,.5E-5,CORLN(1),SL)
       CORML(1) = CORLN(1)/DMDL(1)
       GO TO 90
C
C   NO INITIAL LAMINAR VALUES GIVEN
C   CALCULATE INITIAL CORRELATION NUMBER
C
C   SHARP LEADING EDGE
    70 IF(KLE.NE.1.AND.ABS(DMDL(1)).GE..0001) GO TO 80
       CORLN(1)= 0.
       CORML(1)= 0.
       GO TO 90
C   STAGNATION POINT
    80 CALL CURVFT(CCN,CORLN(1),SW(1),0,5,0)
       CORML(1)= CORLN(1)/DMDL(1)
       IF (CORML(1).LT.0.) GO TO 90
       ERROR= .TRUE.
       WRITE(6,1040)
       RETURN
C
C   SOLVE LAMINAR DIFFERENTIAL EQUATION
C   CALCULATE CORRELATION NUMBERS ALONG THE SURFACE
C
    90 TEM1= 1.+.5*(GAM-1.)*ME(1)**2
       TEM2= (3.*GAM-1.)/(2.*GAM-2.)
       DEL= 0.002*ARCL
       SS= -DEL
       NTAB=1
       CTAB1(1)= CORLN(1)
       CTAB2(1)= CORML(1)
       STAB(1)= 0.
   100 SS= SS+DEL
       SSDEL = SS+DEL
       CALL LGRNGE(S,SW,NST,SS,ANS1)
       CALL LGRNGE(S,ME,NST,SS,ANS2)
       CALL LGRNGE(S,ME,NST,SSDEL,ANS3)
       CALL LGRNGE(S,DMDL,NST,SSDEL,ANS4)
       A1= 0.43631-0.00367*ANS1+0.00481*ANS1**2+0.00651*ANS1**3
       A2= 5.43220+2.25400*ANS1-0.06672*ANS1**2-0.20637*ANS1**3
       A3= 4.51903-10.49775*ANS1-12.71732*ANS1**2-2.95270*ANS1**3
       A4= 19.01831+62.76597*ANS1+115.00986*ANS1**2+62.53113*ANS1**3
       A= A1-A3*CTAB1(NTAB)**2-2.*A4*CTAB1(NTAB)**3
       B= A2+2.*A3*CTAB1(NTAB)+3.*A4*CTAB1(NTAB)**2
```

```
C
C     FOR SW = C.0
      IF ( CTAB1(NTAB).GE.-.1) GO TO 101
      A=.3953
      B=4.739
  101 K1 = 0
      SOL1 = SS/ARCL
      SOL2 = SSDEL/ARCL
      TEM3 = SIMPS1(SOL1,SOL2,INT1,K1)
      IF (TEM3.EQ.0..OR.K1.EQ.0) GO TO 110
      ERROR= .TRUE.
      WRITE(6,1050)
      RETURN
  110 IF (NTAB.GT.1) TEM4= ANS2**(-B)*TEM1**TEM2
      TEM1= 1.+.5*(GAM-1.)*ANS3**2
      TEM5= ANS3**(-B)*TEM1**TEM2
      TEM6= -A*TEM5*TEM3
      IF (NTAB.EQ.1) TEM7=C.
      IF (NTAB.GT.1) TEM7= TEM5/TEM4*CTAB2(NTAB)
      NTAB= NTAB+1
      CTAB2(NTAB)= TEM6+TEM7
      CTAB1(NTAB)= CTAB2(NTAB)*ANS4
      STAB(NTAB)= SSDEL
C
C     WHEN SW IS NOT EQUAL TO 0.0 ,CURVE FIT RANGE ON  CORLN
C     IS FROM -.32 TO .16
C
      IF (CTAB1(NTAB).GT. .16) GO TO 120
      IF (SS.LT.ARCL) GO TO 100
  120 IF (KSDE.NE.1) GO TO 130
      WRITE(6,1060)
      WRITE(6,1070) (STAB(I),CTAB1(I),I=1,NTAB)
C
C  CALCULATE LAMINAR BOUNDARY LAYER PARAMETERS AT EACH STATION
C
  130 IF (KLAM.NE.1) GO TO 140
      WRITE(6,1080)
  140 I= 0
  150 I= I+1
      IF (I.EQ.NTURB) ITRAN=-1
      IF (S(I).LE.STAB(NTAB)) GO TO 160
      ERROR= .TRUE.
      WRITE(6,1090)
      RETURN
  160 CALL LGRNGE(STAB,CTAB1,NTAB,S(I),CORLN(I))
      CALL LGRNGE(STAB,CTAB2,NTAB,S(I),CORML(I))
C  OBTAIN SHEAR, CRN, AND DTH FROM CURVE FITS VS CORLN AND SW
      CALL CURVFT(CSHR,SHEAR(I),CORLN(I),SW(I),3,3)
      CALL CURVFT(CCRN,CRN(I),CORLN(I),SW(I),3,3)
      CALL CURVFT(CDTH,DTH(I),CORLN(I),SW(I),3,3)
C
C     FOR SW = C.0
      IF(CORLN(I).GE. -.1) GO TO 161
      SHEAR(I) = -1.2222*CORLN(I)+.26
      CRN(I) = -58.824 *CORLN(I)-.6765
      DTH(I) = -22.222*CORLN(I)+7.1112
C  CALCULATE OTHER LAMINAR BOUNDARY LAYER PARAMETERS
```

```
161   TEM1= 1.+.5*(GAM-1.)*ME(I)**2
      THET(I)= SQRT(-CORML(I)*NUTZ*SUTHL(I)*ARCL/ATZ*TEM1**((3.-GAM)/
     1(2.*GAM-2.)))
      FORM(I)= (-1.1138*CORLN(I)+2.38411)*(1.+(2.79-1.78*PR**.5)*((1.+
     1SW(I))*TEM1-1.))+(4.65*PR**(1./3.)-3.65*PR**.5)*PR**.5*(TEM1-1.)
      DELSR(I)= THET(I)*FORM(I)
      RTH(I)= UE(I)*THET(I)/NUW(I)
      FORMI(I)= (FORM(I)-SQRT(PR)*(TEM1-1.))/((1.+SW(I))*TEM1)
      FORMTR(I)= FORMI(I)*(1.+SW(I))
      DELTA(I)= THET(I)*(DTH(I)+(TEM1-1.)*(FORMTR(I)+1.))
      SHAPL(I)= DELTA(I)**2/NUW(I)*DUDS(I)
      IF (I.EQ.1) GO TO 180
      CFRW= 2.*SHEAR(I)*SQRT(-SOL(I)/ME(I)/CORML(I))
      CF(I)= CFRW/SQRT(RW(I))
      TAUW(I)= CF(I)*HEADW(I)
      NURW= CFRW*PR**.3/CRN(I)
      NUSS(I)= NURW*SQRT(RW(I))
      DTDY(I)= NUSS(I)*(TAWL(I)-TWAL(I))/S(I)
      HTRAN(I)= TC*DTDY(I)
      IF (TAUW(I).GT.0.) GO TO 180
      IF (KATCH.NE.0) GO TO 170
      ISEP= I
      SBPRN= .TRUE.
      RETURN
  170 ITRAN= -2
      GO TO 270
  180 IF (I.EQ.1.AND.UE(1).EQ.0.) GO TO 190
      SHAPK(I)= NUTZ*RTH(I)**2*SUTHL(I)**2*(1.+SW(I))**4/ATZ/ME(I)**2/
     1FF(I)/ARCL*DMDL(I)*TEM1**(1./(GAM-1.))
      GO TO 200
  190 SHAPK(I)= 0.07
  200 RTHI(I)= RTH(I)*SUTHL(I)*(1.+SW(I))**2/FF(I)/SQRT(TEM1)
C
C  CALCULATE RCRIT TO CHECK FOR INSTABILITY AND TRANSITION
C
      CALL CURVFT(CRCR,RCRIT,SHAPK(I),0,5,0)
      IF (SHAPK(I) .GT. .07) RCRIT = 8.3163
      RCRIT= EXP(RCRIT)
      IF(INST.NE.0) GO TO 210
C
C  CHECK FOR INSTABILITY
C
      IF(RTHI(I).LT.RCRIT) GO TO 270
      RINS= RTHI(I)
      INST= I
      GO TO 270
C
C  CHECK FOR TRANSITION
C
  210 K1= 0
      NS= I
      TEM= SIMPS1(SOL(INST),SOL(I),INT2,K1)
      IF (TEM.EQ.0..OR.K1.EQ.0) GO TO 220
      ERROR= .TRUE.
      WRITE(6,1100)
      RETURN
```

```
  220 KBAR= TEM/(SOL(I)-SOL(INST))
      CALL CURVFT(CDIF,DIFF,KBAR,0,5,0)
      IF (KBAR .GT. .03) DIFF = 44000.*KBAR+700.0
      RTRAN= RINS+DIFF
      IF(RTHI(I).LT.RTRAN) GO TO 270
      IF (I.LT.NTURB) GO TO 270
      ITRAN= -1
      GO TO 270
  230 ITRAN= I
C
C  COMPUTE INITIAL VALUES FOR TURBULENT SOLUTION
C
  240 TRANS= .TRUE.
      IF (DTURB.EQ.0..AND.TTURB.EQ.0.) GO TO 260
      IF (DTURB.GT.0..AND.TTURB.GT.0.) GO TO 250
      ERROR = .TRUE.
      WRITE(6,1110)
      RETURN
  250 THET(ITRAN)= TTURB
      FORM(ITRAN)= DTURB/TTURB
      TEM1= 1.+.5*(GAM-1.)*ME(ITRAN)**2
      FORMI(ITRAN)= (FORM(ITRAN)-PR**(1./3.)*(TEM1-1.))/((1.+SW(ITRAN))
     1*TEM1)
  260 IF (CTHET.GT.0..AND.DTURB.EQ.0..AND.TTURB.EQ.0.) THET(ITRAN)=
     1CTHET*THET(ITRAN)
      THETTR= THET(ITRAN)*(TSE(ITRAN)/TTZ)**((GAM+1.)/(2.*GAM-2.))
      FTRAN= (ME(ITRAN)*ATZ*THETTR/NUTZ)**1.268
      IF (RTRAN.LE.0.) RTRAN=1000.
      FORMS= FORMI(ITRAN)-0.59389-0.06591*ALOG(RTRAN)+0.001272*(ALOG(RTR
     IAN))**2
      IF (DTURB.GT.0..AND.TTURB.GT.0.) FORMS=FORMI(ITRAN)
      RETURN
C
C  PRINT OUTPUT
C
  270 IF (KLAM.NE.1) GO TO 280
      IF(INST.EQ.0 .OR. INST.EQ.I) WRITE(6,1120) I,CORLN(I),SHEAR(I),
     1DTH(I),FORMTR(I),SHAPL(I),RTHI(I),SHAPK(I),RCRIT
      IF(INST.NE.0 .AND. INST.NE.I) WRITE(6,1130) I,CORLN(I),SHEAR(I),
     1DTH(I),FORMTR(I),SHAPL(I),RTHI(I),KBAR,DIFF,RTRAN
      IF (ITRAN.EQ.-2) WRITE(6,1140)
  280 IF(ITRAN.EQ.-1.OR.ITRAN.EQ.-2) GO TO 230
      IF (I.EQ.NST) RETURN
      GO TO 150
C
C  FORMAT STATEMENTS
C
 1000 FORMAT(/////,10X,60HA NEGATIVE INITIAL VALUE HAS BEEN GIVEN. THIS
     1IS NOT ALLOWED)
 1010 FORMAT(/////,10X,75HINITIAL VALUES WERE NOT GIVEN FOR THE TURBULEN
     1T BOUNDARY LAYER AT STATION 1)
 1020 FORMAT(/////,10X,80HINITIAL VALUES WERE GIVEN FOR THE TURBULENT BO
     1UNDARY LAYER AT A STAGNATION POINT)
 1030 FORMAT(/////,10X,94HINITIAL VALUES OTHER THAN ZERO WERE GIVEN FOR
     1THE LAMINAR BOUNDARY LAYER AT A STAGNATION POINT)
```

```
 1040 FORMAT(/////,10X,106HFOR THIS INPUT DATA STATION 1 IS ASSUMED TO B
     1E A STAGNATION POINT, SINCE NO INITIAL THICKNESSES ARE GIVEN./
     210X,118HIN THIS CASE PRESSURE SHOULD DECREASE INITIALLY.  EITHER G
     3IVE AN INITIAL VALUE FOR DISPLACEMENT OR MOMENTUM THICKNESS,/
     410X,60HOR BEGIN WITH A SHORT REGION OF FAVORABLE PRESSURE GRADIENT
     5.)
 1050 FORMAT(/////,10X,37HERROR IN COMPUTING INTEGRAL FOR CORLN)
 1060 FORMAT(1H1///7X,50HLAMINAR DIFFERENTIAL EQUATION - SOLUTION FOR CO
     1RLN///5(24H        S        CORLN      )//)
 1070 FORMAT((5(F12.5,2X,F7.4,3X)))
 1080 FORMAT(1H1///1X,59HLAMINAR CALCULATION OF INSTABILITY AND TRANSITI
     1ON LOCATIONS///1X,7HSTATION,2X,5HCORLN,5X,5HSHEAR,5X,3HDTH,6X,6HFO
     2RMTR,4X,5HSHAPL,9X,4HRTHI,6X,5HSHAPK,9X,5HRCRIT,6X,4HKBAR,10X,4HDI
     3FF,9X,5HRTRAN)
 1090 FORMAT(/////,10X,65HLAMINAR SOLUTION HAS PROCEEDED BEYOND THE RANG
     1E WHERE IT IS VALID)
 1100 FORMAT(/////,10X,36HERROR IN COMPUTING INTEGRAL FOR KBAR)
 1110 FORMAT(/////,10X,64HIF INITIAL TURBULENT VALUES ARE GIVEN, THEY BO
     1TH MUST BE NONZERO)
 1120 FORMAT(I4,1X,5F10.4,1X,F12.1,1X,F10.5,1X,F12.1)
 1130 FORMAT(I4,1X,5F10.4,1X,F12.1,24X,F12.5,1X,F12.1,1X,F12.1)
 1140 FORMAT(/////,10X,85HLAMINAR SEPARATION HAS OCCURRED. ASSUMED TO BE
     1 TRANSITION TO TURBULENT BOUNDARY LAYER)
      END
```

```
$IBFTC TURBL    DECK

      SUBROUTINE TURBLN
      COMMON/C1/GAM,R,PTZ,TTZ,UPMACH,NST,NVP,NTURB,KPVM,KEM,KSMTH,
     1KSPLN,KLE,KATCH,CTHET,DLAM,TLAM,DTURB,TTURB,KPRE,KGRAD,KSDE,KLAM,
     2KMAIN,KPROF,X(100),Y(100),PRES(100),UE(100),ME(100),POPTZ(100),
     3VOVCR(100),TWAL(100)
      COMMON/C2/PSZ,TSZ,UZ,ASZ,ATZ,RHSZ,RHTZ,MUSZ,MUTZ,NUSZ,NUTZ,CP,
     1PR,TC,ARCL
      COMMON/C3/XOM(100),YOM(100),S(100),SOL(100),AE(100),TSE(100),
     1TAWL(100),TAWT(100),TBAR(100),RW(100),SW(100),SUTHL(100),
     2RHSW(100),RHSE(100),HEADW(100),HEADE(100),NUW(100),MUBAR(100),
     3AA(100),BB(100),FF(100),DUDS(100),DMDS(100),DMDL(100)
      COMMON/C4/THET(100),DELSR(100),DELTA(100),FORM(100),
     1FORMI(100),FORMTR(100),RTH(100),RTHI(100),CF(100),
     1TAUW(100),NUSS(100),DTDY(100),HTRAN(100),CRN(100)
      COMMON/C6/FTRAN,FORMS
      COMMON/C7/INST,ITRAN,ISEP
      COMMON/C8/XTAB(505),YTAB1(505),YTAB2(505),NTAB
      COMMON/C9/ERROR,TRANS,SEPRN
      REAL MUSZ,NUSZ,MUTZ,NUTZ,ME,NUW,MUBAR,NUSS
      LOGICAL ERROR,TRANS,SEPRN
C
C  SOLVE TURBULENT BOUNDARY LAYER DIFFERENTIAL EQUATIONS
C  USING RUNGA-KUTTA
      CALL RUNKUT
      IF (KSDE.NE.1) GO TO 10
      WRITE(6,1000)
      WRITE(6,1010) (XTAB(I),YTAB1(I),YTAB2(I),I=1,NTAB)
```

```
C
C  CALCULATE TURBULENT BOUNDARY LAYER PARAMETERS AT EACH STATION
C
   10 DO 30 I=ITRAN,NST
      IF (S(I).LE.XTAB(NTAB)) GO TO 20
      ISEP = I-1
      SEPRN= .TRUE.
      RETURN
   20 TEM1 = 1.+.5*(GAM-1.)*ME(I)**2
      CALL LGRNGE(XTAB,YTAB1,NTAB,S(I),F)
      THETTR= NUTZ*F**.7886/ME(I)/ATZ
      THET(I)= THETTR*(TTZ/TSE(I))**((GAM+1.)/(2.*GAM-2.))
      RTH(I)= UE(I)*THET(I)/NUW(I)
      CALL LGRNGE(XTAB,YTAB2,NTAB,S(I),FORMI(I))
      FORMTR(I)= FORMI(I)*(1.+SW(I))
      FORM(I)= FORMTR(I)*TEM1+PR**(1./3.)*(TEM1-1.)
      DELSR(I)= THET(I)*FORM(I)
      POWER= 2.0/(FORMI(I)-1.0)
      IF (FORMI(I).LT.1.02) POWER=100.
      DELTA(I)= (1.+POWER)*DELSR(I)
      CF(I)= 0.246*EXP(-1.561*FORMI(I))*(UE(I)*THET(I)/NUTZ/(TEM1**(1./(
     1GAM-1.))))**(-.268)*TSE(I)/TBAR(I)*(MUBAR(I)/MUTZ)**(.268)
      TAUW(I)= CF(I)*HEADE(I)
      IF (I.EQ.1) GO TO 30
      HTRAN(I)= CF(I)/2./PR**(2./3.)*RHSE(I)*UE(I)*CP*(TAWT(I)-TWAL(I))
      DTDY(I)= HTRAN(I)/TC
      NUSS(I)= S(I)*DTDY(I)/(TAWT(I)-TWAL(I))
      CRN(I)= CF(I)*RW(I)/NUSS(I)
   30 CONTINUE
      RETURN
 1000 FORMAT(1H1///5X,62HTURBULENT DIFFERENTIAL EQUATIONS - SOLUTION FOR
     1 F  AND  FORMI///4(31H       S          F        FORMI   )//)
 1010 FORMAT((4(F10.5,2X,F8.1,2X,F7.4,2X)))
      END
```

```
$IBFTC PROFI   DECK
      SUBROUTINE PROFIL
      COMMON/C1/GAM,R,PTZ,TTZ,UPMACH,NST,NVP,NTURB,KPVM,KEM,KSMTH,
     1KSPLN,KLE,KATCH,CTHET,DLAM,TLAM,DTURB,TTURB,KPRE,KGRAD,KSDE,KLAM,
     2KMAIN,KPROF,X(100),Y(100),PRES(100),UE(100),ME(100),POPTZ(100),
     3VGVCR(100),TWAL(100)
      COMMON/C3/XOM(100),YOM(100),S(100),SOL(100),AE(100),TSE(100),
     1TAWL(100),TAWT(100),TBAR(100),RW(100),SW(100),SUTHL(100),
     2RHSW(100),RHSE(100),HEADW(100),HEADE(100),NUW(100),MUBAR(100),
     3AA(100),BB(100),FF(100),DUDS(100),DMDS(100),DMDL(100)
      COMMON/C4/THET(100),DELSR(100),DELTA(100),FORM(100),
     1FORMI(100),FORMTR(100),RTH(100),RTHI(100),CF(100),
     1TAUW(100),NUSS(100),DTDY(100),HTRAN(100),CRN(100)
      COMMON/C5/SHAPL(100),SHAPK(100),B,NS
      COMMON/C7/INST,ITRAN,ISEP
      REAL ME,NUSS
C
C  PRINT LOCATIONS OF INSTABILITY, TRANSITION, AND SEPARATION
C
      IF (KMAIN.NE.1) GO TO 60
      WRITE(6,1000)
      IF(INST.EQ.0) GO TO 10
      WRITE(6,1010) INST
      GO TO 20
   10 WRITE(6,1020)
   20 IF (ITRAN.LE.1) GO TO 30
      WRITE(6,1030) ITRAN
      GO TO 40
   30 WRITE(6,1040)
   40 IF(ISEP.EQ.0) GO TO 50
      WRITE(6,1050) ISEP
      GO TO 60
   50 WRITE(6,1060)
C
C  PRINT LOCATIONS OF LAMINAR AND TURBULENT BOUNDARY LAYERS
C
   60 IEND = ITRAN-1
      IF (IEND.EQ.-1.OR.IEND.EQ.0) IEND=ISEP
      IF (IEND.EQ.0) IEND=NST
      IF (KMAIN.NE.1) GO TO 70
      IF (ITRAN.EQ.1) WRITE(6,1070)
      IF (ITRAN.NE.1) WRITE(6,1080) IEND
      IF (ITRAN.EQ.0) WRITE(6,1090)
      IF (ITRAN.EQ.1) WRITE(6,1100) ITRAN,IEND
   70 IF (ITRAN.LE.1) GO TO 80
      IEND = ISEP
      IF(IEND.EQ.0) IEND=NST
      IF (KMAIN.NE.1) GO TO 90
      WRITE(6,1100) ITRAN,IEND
C
C  PRINT CALCULATED BOUNDARY LAYER PARAMETERS
C
   80 IF (KMAIN.NE.1) GO TO 90
      WRITE(6,1110)
      WRITE(6,1120) (I,X(I),S(I),DELSR(I),THET(I),DELTA(I),FORM(I),
     1FORMI(I),I=1,IEND)
      WRITE(6,1130)
      WRITE(6,1140) (I,CF(I),TAUW(I),RTH(I),DTDY(I),NUSS(I),HTRAN(I),
     1CRN(I),I=1,IEND)
```

```
C
C  COMPUTE BOUNDS ON VELOCITY PROFILES
C
   90 IF (KPROF.NE.1) RETURN
      WRITE(6,1150)
      IF(ITRAN.NE.0) GO TO 100
      IL1= 2
      IL2= IENC
      IT1= 0
      IT2= 0
      GO TC 11C
  100 IL1= 2
      IL2= ITRAN-1
      IT1= ITRAN
      IT2= IENC
      IF (IT1.EQ.1) IT1=2
C
C  CALCULATE AND PRINT LAMINAR BOUNDARY LAYER VELOCITY PROFILES
C
  11C NVP1= NVP+1
      IF (IL2.LT.IL1) GO TO 140
      DO 130 I=IL1,IL2
      WRITE(6,1160) I
      AAA= 2.+SHAPL(I)/6.
      BBB= -.5*SHAPL(I)
      CCC= -2.+.5*SHAPL(I)
      DDC= 1.-SHAPL(I)/6.
      DEL= DELTA(I)/FLOAT(NVP)
      YP= -DEL
      DO 120 J=1,NVP1
      YP= YP+DEL
      ETA= YP/DELTA(I)
      YXMAX= YP/X(NST)
      UUE= AAA*ETA+BBB*ETA**2+CCC*ETA**3+DDD*ETA**4
      U= UUE*UE(I)
  120 WRITE(6,1180) ETA,YP,YXMAX,U,UUE
  130 CONTINUE
C
C  CALCULATE AND PRINT TURBULENT BOUNDARY LAYER VELOCITY PROFILES
C
  140 IF(IT1.EQ.0) RETURN
      DO 160 I=IT1,IT2
      POWER= DELTA(I)/DELSR(I)-1.
      WRITE(6,1170) I,POWER
      DEL= DELTA(I)/FLOAT(NVP)
      YP= -DEL
      DO 150 J=1,NVP1
      YP= YP+DEL
      ETA= YP/DELTA(I)
      YXMAX= YP/X(NST)
      UUE= ETA**(1./POWER)
      U= UUE*UE(I)
  150 WRITE(6,1180) ETA,YP,YXMAX,U,UUE
  160 CONTINUE
      RETURN
```

```
C
C  FORMAT STATEMENTS
C
 1000 FORMAT(1H1///1X,36HPRINCIPAL BOUNDARY LAYER INFORMATION///)
 1010 FORMAT (/10X,31HINSTABILITY OCCURS AT STATION  ,I3)
 1020 FORMAT (/10X,26HINSTABILITY DOES NOT OCCUR)
 1030 FORMAT (/10X,30HTRANSITION OCCURS AT STATION  ,I3)
 1040 FORMAT (/10X,25HTRANSITION DOES NOT OCCUR)
 1050 FORMAT (/10X,30HSEPARATION OCCURS AT STATION  ,I3)
 1060 FORMAT (/10X,25HSEPARATION DOES NOT OCCUR)
 1070 FORMAT (/10X,37HLAMINAR BOUNDARY LAYER DOES NOT OCCUR)
 1080 FORMAT (/10X,42HLAMINAR BOUNDARY LAYER - STATIONS  1  TO  ,I3)
 1090 FORMAT (/10X,39HTURBULENT BOUNDARY LAYER DOES NOT OCCUR///)
 1100 FORMAT (/10X,35HTURBULENT BOUNDARY LAYER - STATIONS,2X,
     1I3,6H  TO  ,I3///)
 1110 FORMAT(/1X,7HSTATION,8X,1HX,12X,1HS,12X,5HDELSR,10X,4HTHET,11X,
     15HDELTA,11X,4HFORM,10X,5HFORMI)
 1120 FORMAT(2X,I3,3X,2F13.6,F14.6,1X,F14.6,1X,F14.6,1X,2F14.4)
 1130 FORMAT(///1X,7HSTATION,6X,2HCF,13X,4HTAUW,11X,3HRTH,14X,4HDTDY,
     113X,4HNUSS,10X,5HHTRAN,12X,3HCRN)
 1140 FORMAT(I5,F14.5,2X,F14.5,1X,F12.1,5X,F14.2,2X,F14.2,1X,
     1F14.4,2X,F13.3)
 1150 FORMAT(1H1///1X,17HVELOCITY PROFILES///)
 1160 FORMAT (/1X,7HSTATION,1X,I5,2X,7HPROFILE/3X,7HY/DELTA,9X,
     11HY,12X,6HY/XMAX,10X,1HU,12X,4HU/UE)
 1170 FORMAT (/1X,7HSTATION,1X,I5,2X,7HPROFILE,28X,2HN=,1X,F6.2/3X,7HY/D
     1ELTA,9X,1HY,12X,6HY/XMAX,10X,1HU,12X,4HU/UE)
 1180 FORMAT(1X,F8.4,2X,2G15.6,2X,F9.2,6X,F8.4)
      END




$IBFTC RUNKU   DECK

      SUBROUTINE RUNKUT
C
C  RUNKUT SOLVES SIMULTANEOUS FIRST ORDER INITIAL VALUE
C  ORDINARY DIFFERENTIAL EQUATIONS
C
      COMMON/C1/GAM,R,PTZ,TTZ,UPMACH,NST,NVP,NTURB,KPVM,KEM,KSMTH,
     1KSPLN,KLE,KATCH,CTHET,DLAM,TLAM,DTURB,TTURB,KPRE,KGRAD,KSDE,KLAM,
     2KMAIN,KPROF,X(100),Y(100),PRES(100),UE(100),ME(100),POPTZ(100),
     3VOVCR(100),TWAL(100)
      COMMON/C3/XOM(100),YOM(100),S(100),SOL(100),AE(100),TSE(100),
     1TAWL(100),TAWT(100),TBAR(100),RW(100),SW(100),SUTHL(100),
     2RHSW(100),RHSE(100),HEADW(100),HEADE(100),NUW(100),MUBAR(100),
     3AA(100),BB(100),FF(100),DUDS(100),DMDS(100),DMDL(100)
      COMMON/C6/FTRAN,FORMS
      COMMON/C7/INST,ITRAN,ISEP
      COMMON/C8/XTAB(505),YTAB1(505),YTAB2(505),NTAB
      DIMENSION YY(2),RY(2),YINC(2),DOT(2),RUK(2,4)
      DOUBLE PRECISION XX,RX,YY,RY,RUK,DEL,DOT,
     1TEM1,TEM2,TEM3,TEM4,TEM5,TEM6
      REAL ME,NUW,MUBAR
```

```
C
C  SET DEL SPACING AND STORE INITIAL VALUES
C
      DEL= 0.002*S(NST)
   10 YY(1)=FTRAN
      YY(2)= FCRMS
      XX= S(ITRAN)
      NV=2
      NTAB = 1
      YTAB1(1)= YY(1)
      YTAB2(1)= YY(2)
      XTAB(1)= XX
C
C  SOLVE FOR YY(1) AND YY(2) AT NEXT XX INCREMENT
C
C  SAVE PREVIOUS YY(1) AND YY(2)
   20 DO 30 J=1,NV
   30 RY(J)= YY(J)
      RX= XX
C
C  CALCULATE NEW YY(1) AND YY(2)
C
      DO 90 L=1,4
C  PUT DIFFERENTIAL EQUATIONS IN THE FORM OF
C  FIRST DERIVATIVE = REMAINDER OF EQUATION
      CALL LGRNGE(S,ME,NST,XX,ANS1)
      CALL LGRNGE(S,SW,NST,XX,ANS2)
      CALL LGRNGE(S,AA,NST,XX,ANS3)
      CALL LGRNGE(S,BB,NST,XX,ANS4)
      CALL LGRNGE(S,DMDS,NST,XX,ANS5)
      CALL LGRNGE(S,TBAR,NST,XX,ANS6)
      TEM1= 1.+(1.+ANS2)*YY(2)
      TEM2= .123*EXP(-1.561*YY(2))*ANS3
      DOT(1)= 1.268*(-YY(1)/ANS1*ANS5*TEM1+TEM2)
      TEM3= YY(2)*(YY(2)+1.)**2*(YY(2)-1.)
      TEM4= 1.+ANS2*(YY(2)*YY(2)+4.*YY(2)-1.)/((YY(2)+1.)*(YY(2)+3.))
      TEM5= (YY(2)*YY(2)-1.)*YY(2)/YY(1)*(.123*EXP(-1.561*YY(2))*ANS3)
      TEM6= (YY(2)*YY(2)-1.)/YY(1)**(.7886)*(.011*(YY(2)+1.)*(YY(2)-1.)
     1**2/YY(2)**2*TTZ/ANS6)*ANS4
      DOT(2)= -ANS5*.5/ANS1*TEM3*TEM4+TEM5-TEM6
C  APPLY THE RUNGA-KUTTA SCHEME
      DO 40 J=1,NV
   40 RUK(J,L) = DEL*DOT(J)
      GO TO (50,50,70,90), L
   50 DO 60 J=1,NV
   60 YY(J)= RY(J)+RUK(J,L)/2.
      XX= RX+DEL/2.
      GO TO 90
   70 DO 80 J=1,NV
   80 YY(J)= RY(J)+RUK(J,L)
      XX= RX+DEL
   90 CONTINUE
C  INCREMENT THE DEPENDENT VARIABLES TO OBTAIN NEW YY(1) AND YY(2)
      DO 100 J=1,NV
      YINC(J) = (RUK(J,1)+2.*RUK(J,2)+2.*RUK(J,3)+RUK(J,4))/6.
  100 YY(J)= RY(J)+YINC(J)
      IF (YY(2).GT.2.8) RETURN
```

```
C
C  STORE NEW CCMPUTED VALUES IN A TABLE
C
      NTAB = NTAB+1
      YTAB1(NTAB)= YY(1)
      YTAB2(NTAB)= YY(2)
      XTAB(NTAB)= XX
      IF (XX.LT.S(NST)) GO TO 20
      RETURN
      END
```

```
$IBFTC FUNC     DECK
      SUBROUTINE FUNCT(XX,FX,DFX,INF)
      COMMON/C1/GAM,R,PTZ,TTZ,UPMACH,NST,NVP,NTURB,KPVM,KEM,KSMTH,
     1KSPLN,KLE,KATCH,CTHET,DLAM,TLAM,DTURB,TTURB,KPRE,KGRAD,KSDE,KLAM,
     2KMAIN,KPROF,X(100),Y(100),PRES(100),UE(100),ME(100),POPTZ(100),
     3VOVCR(10C),TWAL(100)
      COMMCN/C2/PSZ,TSZ,UZ,ASZ,ATZ,RHSZ,RHTZ,MUSZ,MUTZ,NUSZ,NUTZ,CP,
     1PR,TC,ARCL
      COMMON/C3/XOM(100),YOM(100),S(100),SOL(100),AE(100),TSE(100),
     1TAWL(100),TAWT(100),TBAR(100),RW(100),SW(100),SUTHL(100),
     2RHSW(100),RHSE(100),HEADW(100),HEADE(100),NUW(100),MUBAR(100),
     3AA(100),BB(100),FF(100),DUDS(100),DMDS(100),DMDL(100)
      REAL MUSZ,NUSZ,MUTZ,NUTZ,ME,NUW,MUBAR
      INF = 0
      B1= 1.+.5*(GAM-1.)*ME(1)**2
      B2= 1.+(2.79-1.78*PR**.5)*((1.+SW(1))*B1-1.)
      B3= -NUTZ*SUTHL(1)*ARCL/ATZ/DMDL(1)*B1**((3.-GAM)/(2.*GAM-2.))
      B4= -1.1138*B2
      B5= 2.38411*B2+(4.65*PR**(1./3.)-3.65*PR**.5)*PR**.5*(B1-1.)
      FX= (B3*XX)**.5*(B4*XX+B5)
      IF (XX.EQ.0.) GO TO 10
      DFX= .5*(B3*XX)**(-.5)*B3*(B4*XX+B5)+B4*(B3*XX)**.5
      RETURN
   10 INF = 1
      DFX = 1.E10
      RETURN
      END
```

```
$IBFTC INTG1    DECK
      REAL FUNCTION INT1(XX)
      COMMON/C1/GAM,R,PTZ,TTZ,UPMACH,NST,NVP,NTURB,KPVM,KEM,KSMTH,
     1KSPLN,KLE,KATCH,CTHET,DLAM,TLAM,DTURB,TTURB,KPRE,KGRAD,KSDE,KLAM,
     2KMAIN,KPROF,X(100),Y(100),PRES(100),UE(100),ME(100),POPTZ(100),
     3VOVCR(100),TWAL(100)
      COMMON/C3/XOM(100),YOM(100),S(100),SOL(100),AE(100),TSE(100),
     1TAWL(100),TAWT(100),TBAR(100),RW(100),SW(100),SUTHL(100),
     2RHSW(100),RHSE(100),HEADW(100),HEADE(100),NUW(100),MUBAR(100),
     3AA(100),BB(100),FF(100),DUDS(100),DMDS(100),DMDL(100)
      COMMON/C5/SHAPL(100),SHAPK(100),B,NS
      REAL ME,NUW,MUBAR,INT1
      CALL LGRNGE(SOL,ME,NST,XX,ANS)
      INT1= ANS**(B-1.)/((1.+.5*(GAM-1.)*ANS**2)**
     1((3.*GAM-1.)/(2.*GAM-2.)))
      RETURN
      END
```

```
$IBFTC INTG2    DECK
      REAL FUNCTION INT2(XX)
      COMMON/C1/GAM,R,PTZ,TTZ,UPMACH,NST,NVP,NTURB,KPVM,KEM,KSMTH,
     1KSPLN,KLE,KATCH,CTHET,DLAM,TLAM,DTURB,TTURB,KPRE,KGRAD,KSDE,KLAM,
     2KMAIN,KPROF,X(100),Y(100),PRES(100),UE(100),ME(100),POPTZ(100),
     3VOVCR(100),TWAL(100)
      COMMON/C3/XOM(100),YOM(100),S(100),SOL(100),AE(100),TSE(100),
     1TAWL(100),TAWT(100),TBAR(100),RW(100),SW(100),SUTHL(100),
     2RHSW(100),RHSE(100),HEADW(100),HEADE(100),NUW(100),MUBAR(100),
     3AA(100),BB(100),FF(100),DUDS(100),DMDS(100),DMDL(100)
      COMMON/C5/SHAPL(100),SHAPK(100),B,NS
      REAL ME,NUW,MUBAR,INT2
      IF (NS.LT.4) GO TO 10
      CALL LGRNGE(SOL,SHAPK,NS,XX,INT2)
      RETURN
   10 DO 20 J=2,NS
      IF (SOL(J).LT.XX) GO TO 20
      INT2= SHAPK(J-1)+(SHAPK(J)-SHAPK(J-1))*(XX-SOL(J-1))/(SOL(J)-SOL(J
     1-1))
      RETURN
   20 CONTINUE
      RETURN
      END
```

```
$IBFTC SPLIN    DECK
      SUBROUTINE SPLINE(X,Y,N,DYDX,D2YDX2)
C
C  SPLINE FITS A SPLINE CURVE TO  X  AND  Y
C  AND CALCULATES FIRST AND SECOND DERIVATIVES AT THE SPLINE POINTS
C  END POINT SECOND DERIVATIVES EQUAL THOSE AT ADJACENT POINTS
C
      DIMENSION X(N),Y(N),DYDX(N),D2YDX2(N)
      DIMENSION G(100),H(100)
      G(1)= -1.
      H(1)= 0.
      N1= N-1
      IF (N1.LT.2) GO TO 20
      DO 10 I=2,N1
      A= (X(I)-X(I-1))/6.
      B= (X(I+1)-X(I))/6.
      C= 2.*(A+B)-A*G(I-1)
      D= (Y(I+1)-Y(I))/(X(I+1)-X(I))-(Y(I)-Y(I-1))/(X(I)-X(I-1))
      G(I)= B/C
   10 H(I)= (D-A*H(I-1))/C
   20 D2YDX2(N)= H(N1)/(1.+G(N1))
      DO 30 I=2,N
      K= N+1-I
   30 D2YDX2(K)= H(K)-G(K)*D2YDX2(K+1)
      DYDX(1)= (X(1)-X(2))/6.*(2.*D2YDX2(1)+D2YDX2(2))+(Y(2)-Y(1))/(X(2)
     1-X(1))
      DO 40 I=2,N
   40 DYDX(I)= (X(I)-X(I-1))/6.*(2.*D2YDX2(I)+D2YDX2(I-1))+(Y(I)-Y(I-1))
     1/(X(I)-X(I-1))
      RETURN
      END
```

```
$IBFTC ROO      DECK
      SUBROUTINE ROOT(A,B,Y,FUNCT,TOLERY,X,DFX)
C
C  ROOT FINDS A ROOT FOR (FUNCT-Y) IN THE INTERVAL (A,B)
C
      X1= A
      X2= B
      CALL FUNCT(X1,FX1,DFX,INF)
   10 DO 30 I=1,20
      X= (X1+X2)/2.
      CALL FUNCT(X,FX,DFX,INF)
      IF ((FX1-Y)*(FX-Y).GT.0.) GO TO 20
      X2= X
      GO TO 30
   20 X1= X
      FX1= FX
   30 CONTINUE
      IF (ABS(Y-FX).LT.TOLERY) RETURN
      WRITE(6,1000) A,B,Y
      STOP
 1000 FORMAT(////4X,49HROOT HAS FAILED TO CONVERGE IN THE GIVEN INTERVAL
     1/4X,3HA =,G14.6,10X,3HB =,G14.6,10X,3HY =,G14.6)
      END
```

```
$IBFTC LGRNG    DECK
      SUBROUTINE LGRNGE(X,Y,N,ARG,ANS)
C
C  LGRNGE PERFORMS 4 POINT LAGRANGIAN INTERPOLATION
C
      DIMENSION X(N),Y(N),XX(4),YY(4)
      IF (ARG-X(2)) 10,10,20
   10 MM = 1
      GO TO 70
   20 IF (ARG-X(N-1)) 40,40,30
   30 MM = N-3
      GO TO 70
   40 N1 = N-1
      DO 60 I=2,N1
      IF (ARG-X(I)) 50,50,60
   50 MM = I-2
      GO TO 70
   60 CONTINUE
   70 DO 80 I=1,4
      MMM = MM+I-1
      XX(I) = X(MMM)
   80 YY(I) = Y(MMM)
      C1 = ((ARG-XX(2))*(ARG-XX(3))*(ARG-XX(4)))/
     1(XX(1)-XX(2))/(XX(1)-XX(3))/(XX(1)-XX(4))
      C2 = ((ARG-XX(1))*(ARG-XX(3))*(ARG-XX(4)))/
     1(XX(2)-XX(1))/(XX(2)-XX(3))/(XX(2)-XX(4))
      C3 = ((ARG-XX(1))*(ARG-XX(2))*(ARG-XX(4)))/
     1(XX(3)-XX(1))/(XX(3)-XX(2))/(XX(3)-XX(4))
      C4 = ((ARG-XX(1))*(ARG-XX(2))*(ARG-XX(3)))/
     1(XX(4)-XX(1))/(XX(4)-XX(2))/(XX(4)-XX(3))
      ANS = C1*YY(1)+C2*YY(2)+C3*YY(3)+C4*YY(4)
      RETURN
      END
```

```
$IBFTC CURVF    DECK

      SUBROUTINE CURVFT(COEF,ANS,X,Y,NX,NY)
C
C  EVALUATE THE POLYNOMIAL FUNCTION, ANS=F(X,Y), USING COEFFICIENTS, COEF
C
      DIMENSION COEF(20)
      NX1 = NX+1
      NY1 = NY+1
      ANS = COEF(1)
      IF (X.EQ..0.AND.Y.EQ..0) RETURN
      IF (Y.EQ..0) GO TO 10
      IF (X.EQ..0) GO TO 30
      GO TO 50
   10 DO 20 I=2,NX1
   20 ANS = ANS+COEF(I)*X**(I-1)
      RETURN
   30 DO 40 I=2,NY1
      K = (I-1)*NX1+1
   40 ANS = ANS+COEF(K)*Y**(I-1)
      RETURN
   50 ANS = .0
      DO 60 I=1,NY1
      DO 60 J=1,NX1
      K = (I-1)*NX1+J
   60 ANS = ANS+COEF(K)*Y**(I-1)*X**(J-1)
      RETURN
      END




$IBFTC SIMP     DECK

      FUNCTION SIMPS1(X1,X2,FUNC,KSIG)
      DIMENSION V(200),H(200),A(200),B(200),C(200),P(200),E(200)
      LOGICAL SPILL
      DOUBLE PRECISION ANS,Q
      DATA TWO,THREE,FOUR,THIRTY/2.0,3.0,4.0,30.0/
      DATA T,NMAX,NSIG/3.0E-5,200,1/
C  INITIALIZE FIRST ELEMENTS OF ARRAYS.
      V=X1
      H=(X2-V)/TWO
      A=FUNC(V)
      B=FUNC(V+H)
      C=FUNC(X2)
      P=H*(A+FOUR*B+C)
      E=P
      ANS=P
      N=1
      FRAC=T
      SPILL=.FALSE.
   10 TEST=ABS(FRAC*ANS)
      K=N
      DO 30 I=1,K
C  TEST MAGNITUDE OF 4TH ORDER ERROR IN THIS INTERVAL.
      IF (ABS(E(I)).LE.TEST) GO TO 30
      IF (N.LT.NMAX) GO TO 20
```

```
C  GC TO FINISH IF STORAGE IS FILLED UP.
      SPILL=.TRUE.
      KSIG=KSIG+NSIG
      GO TC 40
C  SUBDIVICE INTERVAL AGAIN TO REDUCE 4TH ORDER ERROR.
   2C N=N+1
      V(N)=V(I)+H(I)
      H(N)=H(I)/TWO
      A(N)=B(I)
      B(N)=FUNC(V(N)+H(N))
      C(N)=C(I)
      P(N)=H(N)*(A(N)+FOUR*B(N)+C(N))
      H(I)=H(N)
      B(I)=FUNC(V(I)+H(I))
      C(I)=A(N)
      Q=P(I)
      P(I)=H(I)*(A(I)+FOUR*B(I)+C(I))
      Q=P(I)+P(N)-Q
      ANS=ANS+C
      E(I)=C
      E(N)=Q
   3C CONTINUE
C  TEST ALL INTERVALS AGAIN IF ANY WERE SUBDIVIDED THE LAST TIME.
      IF (N.GT.K) GO TO 10
   40 Q=0.C
      DO 5C I=1,N
   50 Q=C+E(I)
C  TIGHTEN ERRCR LIMIT IF TOTAL ACCUMULATED ERROR TOO LARGE.
      IF (ABS(C/T).LE.ABS(ANS).OR.SPILL) GO TO 60
      FRAC=FRAC/TWO
      GO TC 10
C  FINISH CALCLLATION.
   60 SIMPS1=(ANS+Q/THIRTY)/THREE
      RETURN
      ENC
```

```
$IBFTC AFTMIX  DECK
      SUBROUTINE AFMIX(ALPH1,TE,SP,XMFS1,NP)
      DIMENSION XXX(100),YYY(100),SSS(100)
      COMMON/C1/GAM,R,PTZ,TTZ,UPMACH,NST,NVP,NTURB,KPVM,KEM,KSMTH,
     1KSPLN,KLE,KATCH,CTHET,DLAM,TLAM,DTURB,TTURB,KPRE,KGRAD,KSDE,KLAM,
     2KMAIN,KPROF,X(100),Y(100),PRES(100),UE(100),ME(100),POPTZ(100),
     3VOVCR(100),TWAL(100)
      COMMON/C3/XOM(100),YOM(100),S(100),SOL(100),AE(100),TSE(100),
     1TAWL(100),TAWT(100),TBAR(100),RW(100),SW(100),SUTHL(100),
     2RHSW(100),RHSE(100),HEADW(100),HEADE(100),NUW(100),MUBAR(100),
     3AA(100),BB(100),FF(100),DUDS(100),DMDS(100),DMDL(100)
      COMMON/C4/THET(100),DELSR(100),DELTA(100),FORM(100),
     1FORMI(100),FORMTR(100),RTH(100),RTHI(100),CF(100),
     1TAUW(100),NUSS(100),DTDY(100),HTRAN(100),CRN(100)
      EQUIVALENCE (X,XXX),(Y,YYY),(S,SSS),(NST,N)
      REAL ME
      KODE = 0
      DELS = DELSR(N)
      DELP = DELSR(NP)
      THETS = THET(N)
      THETP = THET(NP)
      WRITE(6,5)
5     FORMAT(1H1,20X,22HAFTERMIXING PROPERTIES)
      WRITE(6,1)
1     FORMAT(1H0,10X,40HNOZZLE WITH NO BOUNDARY LAYER CORRECTION)
2     VVCR1=SQRT(((GAM+1.0)/2.0*XMFS1**2)/(1.0+(GAM-1.0)/2.0*XMFS1**2))
      XX=SP*COS(ALPH1)
      DELSRT= (DELS+DELP)/XX
      THETA= (THETS+THETP)/XX
      DTE = TE/XX
      AFS1 = (GAM-1.0)/(GAM+1.0)*VVCR1**2
      A = 1.0-DELSRT-DTE-THETA
      A1= 1.0-DELSRT-DTE
      IF (A  .LE. 0.0) GO TO 16
      C=((1.0-AFS1)*(GAM+1.0)/(2.0*GAM)+(COS(ALPH1))**2*
     1A*VVCR1**2)/(COS(ALPH1)*A1*VVCR1)
      D = VVCR1*SIN(ALPH1)*A/A1
       VXVCR2 = GAM*C/(GAM+1.0)-SQRT((GAM*C/(GAM+1.0))**2
     1-1.0+(GAM-1.0)/(GAM+1.0)*D**2)
      GO TO 4
3     KODE=2
      WRITE(6,10)
10    FORMAT(1H0,10X,19HSUPERSONIC SOLUTION)
       VXVCR2 = GAM*C/(GAM+1.0)+SQRT((GAM*C/(GAM+1.0))**2
     1-1.0+(GAM-1.0)/(GAM+1.0)*D**2)
4     BENCH =      (GAM-1.0)/(GAM+1.0)*(D**2+VXVCR2**2)
      IF (BENCH.GE. 1.0) RETURN
      DENSR2= (1.0-(GAM-1.0)/(GAM+1.0)*(D**2+VXVCR2**2))
     1**(1.0/(GAM-1.0))
      DENSR1 = 1.0/((1.0+(GAM-1.0)/2.0*XMFS1**2)**(1.0/(
     1GAM-1.0)))
      PR2 = DENSR2**GAM
      PT2PTO=(DENSR1*VVCR1*COS(ALPH1)*A1)/(DENSR2*VXVCR2)
      PTOP2 = 1.0/(PT2PTO*PR2)
      EBAR2 = ((PT2PTO)**((1.0-GAM)/GAM)-1.0)/(PTOP2**
     1((GAM-1.0)/GAM)-1.0)
      ETAN = 1.0-EBAR2
      VVCR2 = SQRT(D**2+VXVCR2**2)
      XM2 = SQRT(((2.0/(GAM+1.0))*VVCR2**2)/(1.0-((GAM-1.0)/(GAM+1.0))
```

```
     1*VVCR2**2))
      T2TTO = 1.0 - (GAM-1.)/(GAM+1.)*VVCR2**2
      ALPH2 = ATAN(D/VXVCR2)
      XMX1 = XMFS1*COS(ALPH1)
      XMX2 = XM2*COS(ALPH2)
      ALPH2 = ALPH2*57.2958
      ALPH1 = ALPH1*57.2958
      WRITE(6,6) XMFS1,SP,TE,XM2,VVCR1,XMX1,XMX2
6     FORMAT(1H0,7HXMFS1 =,F6.4,2X,9HSPACING =,F7.6,2X,4HTE =,F7.5,2X,
     15HXM2 =,F6.4,2X,8HV/VCR1 =,F6.3,2X,6HXMX1 =,F6.3,2X,6HXMX2 =,F6.3)
      WRITE(6,7)ALPH1,ALPH2,PT2PTO,PTOP2,T2TTO,VVCR2,EBAR2,ETAN
7     FORMAT(1H0,6HALPH1=,F7.3,2X,6HALPH2=,F7.3,2X,8HPT2/PTO=,F7.4,2X,
     17HPTO/P2=,F9.3,2X,7HT2/TTO=,F7.4,2X,7HV/VCR2=,F6.3,2X,6HEBAR2=,
     2F7.5,2X,6HETA-N=,F6.4)
      IF (KODE .EQ. 2 ) RETURN
      IF (KODE.EQ.1 ) WRITE(6,15) DELSR(N+1),DELSR(N+2),THET(N+1),
     1THET(N+2)
15    FORMAT(1H0,12HDELSR(N+1) =,F8.6,2X,12HDELSR(N+2) =,F8.6,2X,
     111HTHET(N+1) =,F8.6,2X,11HTHET(N+2) =,F8.6)
      IF (KODE .EQ. 1 .AND. XMX1 .LT. 1.0000 ) RETURN
      ALPH1 = ALPH1*.017453
      IF( KODE.EQ.1 ) GO TO 3
16    SP= 2.0*(YYY(NP)+DELP)/COS(ALPH1)
C
C    DISPLACEMENT THICKNESS HAS BEEN ADDED TO THE NOZZLE EXIT
C
C    INCREASING THE STRAIGHT SECTION
      Z = 2.0*(YYY(NP)+DELP)*TAN(ALPH1)
      DELZ = Z-(XXX(N)-XXX(NP))
      XXX(N+1) = DELZ+XXX(N)
      YYY(N+1) = YYY(N)
C
C    FURTHER INCREASING OF THE STRAIGHT SECTION  DUE TO INCREASING
C    BOUNDARY LAYER
C
C    NOZZLE ANGLE IS CONSTANT
      FA = ATAN((DELSR(N)-DELSR(N-2))/(SSS(N)-SSS(N-2)))
      DELSR(N+1) = DELSR(N)+DELZ*TAN(FA)
      FAMT = ATAN((THET(N)-THET(N-2))/(SSS(N)-SSS(N-2)))
      THET(N+1) = THET(N)+DELZ*TAN(FAMT)
      ALPW = 3.14159 - ALPH1 - (3.14159/2.0 +FA)
      DS = SIN(3.14159/2.0+FA)*(DELSR(N+1)-DELSR(NP))/SIN(ALPW)
      DF = SIN(ALPH1)*(DELSR(N+1)-DELSR(NP))/SIN(ALPW)
      DELZF = DF*COS(FA)
      DELSR(N+2) = DELSR(N+1) + DELZF*TAN(FA)
      THET(N+2) = THET(N+1) + DELZF*TAN(FAMT)
      XXX(N+2) = XXX(N+1) + DELZF
      YYY(N+2) = YYY(N)
      SP = SP +DS
      DELS = DELSR(N+2)
      THETS = THET(N+2)
      WRITE(6,20)
20    FORMAT(1H0,10X,37HNOZZLE WITH BOUNDARY LAYER CORRECTION)
      KODE = 1
      GO TO 2
      END
```

```
$IBFTC NCZZC    DECK
      SUBROUTINE NOZZLC(NSP,ALPH1,NCP)
      DIMENSION YYC(100),YYCL(100),XXT(100),YYT(100),XXTL(100),YYTL(100)
      DIMENSION XXX(100),YYY(100)
      COMMON/C1/GAM,R,PTZ,TTZ,UPMACH,NST,NVP,NTURB,KPVM,KEM,KSMTH,
     1KSPLN,KLE,KATCH,CTHET,DLAM,TLAM,DTURB,TTURB,KPRE,KGRAD,KSDE,KLAM,
     2KMAIN,KPROF,X(100),Y(100),PRES(100),UE(100),ME(100),POPTZ(100),
     3VOVCR(100),TWAL(100)
      COMMON/C4/THET(100),DELSR(100),DELTA(100),FORM(100),
     1FORMI(100),FORMTR(100),RTH(100),RTHI(100),CF(100),
     1TAUW(100),NUSS(100),DTDY(100),HTRAN(100),CRN(100)
      EQUIVALENCE (X,XXX),(Y,YYY),(NST,N)
      N1 = N+2
      XSUB = XXX(NSP)
      DO 10 I = NSP,N1
      XXX(I) =XXX(I) -XSUB
      YYC(I) = YYY(I)+DELSR(I)
      IF (I.GT.NCP) GO TO 10
      YYCL(I) = -YYC(I)
10    CONTINUE
      ALPH1 = ALPH1*.017453
      DO 11 I=NSP,N1
      XXT(I) = XXX(I)*COS(ALPH1)- YYC(I)*SIN(ALPH1)
      YYT(I) =  YYC(I)*COS(ALPH1) +XXX(I)*SIN(ALPH1)
      IF (I.GT.NCP) GO TO 11
      XXTL(I)= XXX(I)*COS(ALPH1)-YYCL(I)*SIN(ALPH1)
      YYTL(I)=YYCL(I) *COS(ALPH1) +XXX(I)*SIN(ALPH1)
11    CONTINUE
      WRITE(6,1)
1     FORMAT(1H1,20X,18HNOZZLE COORDINATES,1X,18HSUPERSONIC SECTION)
      WRITE(6,2)
2     FORMAT(1H0,10X,37HNOZZLE WITH BOUNDARY LAYER CORRECTION)
      WRITE(6,3)
3     FORMAT(1H0,9X,1HX,11X,1HY,10X,3HXTU,9X,3HYTU,8X,3HXTL,10X,3HYTL)
      DO 12 I=NSP,N1
12    WRITE(6,4)XXX(I),YYC(I),XXT(I),YYT(I),XXTL(I),YYTL(I)
4     FORMAT(1H ,6F12.6)
      RETURN
      END
```

Lewis Research Center,
National Aeronautics and Space Administration,
Cleveland, Ohio, May 19, 1971,
120-34.

APPENDIX A

PROGRAM CHANGES FOR A GAS OTHER THAN AIR

The program gas properties are set up for air. This program can be easily changed so that it applies to gases other than air. The changes are all made in subroutine PRECAL.

The coefficients read in by the DATA statements for CMU, CPR, and CTC arrays must be changed. The equations for the curve fits have the following form:

$$\frac{\mu}{\mu_{sl}} = a + b\left(\frac{T}{T_{sl}}\right) + c\left(\frac{T}{T_{sl}}\right)^2 + d\left(\frac{T}{T_{sl}}\right)^3 + e\left(\frac{T}{T_{sl}}\right)^4$$

$$Pr = a_1 + b_1\left(\frac{T}{T_{sl}}\right) + c_1\left(\frac{T}{T_{sl}}\right)^2 + d_1\left(\frac{T}{T_{sl}}\right)^3 + e_1\left(\frac{T}{T_{sl}}\right)^4$$

$$\frac{k}{k_{sl}} = a_2 + b_2\left(\frac{T}{T_{sl}}\right) + c_2\left(\frac{T}{T_{sl}}\right)^2 + d_2\left(\frac{T}{T_{sl}}\right)^3 + e_2\left(\frac{T}{T_{sl}}\right)^4$$

If the number of coefficients changes from five in any case, this must be reflected both in the DATA statements and later in the calls on CURVFT where CMU, CPR and CTC are used. If the properties are put in a different form, these cards must be removed.

The sea-level reference values in U.S. Customary and SI Units of temperature (TSLE, TSLM), viscosity (MUSLE, MUSLM), and thermal conductivity (TCSLE, TCSLM) must be changed.

The value of Sutherland's constant (TCON) and the computation of k_{su} (SUTHL(I)) will have to be changed. A temperature-viscosity law of the following form was used:

$$\frac{\mu}{\mu_o} = k_{su}\frac{T}{T_o}$$

where for air

$$k_{su} = \left(\frac{T}{T_o}\right)^{1/2}\left(\frac{T_o + TCON}{T + TCON}\right)$$

with TCON being Sutherland's constant for air.

APPENDIX B

ADDITIONAL OUTPUT

This output is obtained when KPRE, KGRAD, KSDE, KLAM, and KPROF are set equal to 1.

The output corresponding to KPRE are the geometric variables

X	X-coordinate, m (ft)
Y	Y-coordinate, m (ft)
S	surface length, x, m (ft)
XOM	ratio of X to maximum X-coordinate
YOM	ratio of Y to maximum X-coordinate
SOL	ratio of surface length to total arc length

The next part of this output gives the local speed of sound and the several temperatures

AE	local free-stream speed of sound, m/sec (ft/sec)
TSE	static temperature, K (^{o}R)
TWAL	wall temperature, K (^{o}R)
TAWL	laminar recovery temperature, K (^{o}R)
TAWT	turbulent recovery temperature, K (^{o}R)
TBAR	reference temperature, K (^{o}R)

The final part of this output gives

RW	Reynolds number at wall, RW = (UE)(S)/NUW
SW	temperature function at wall
SUTHL	value of coefficient in Sutherland's viscosity temperature formula
RHSW	static density based on wall temperature, kg/m^3 (slug/ft^3)
RHSE	static density based on free-stream temperature, kg/m^3 (slug/ft^3)
HEADW	velocity head based on density at wall, N/m^2 (lbf/ft^2)
HEADE	velocity head based on free-stream density, N/m^2 (lbf/ft^2)
NUW	kinematic viscosity at wall, m^2/sec (ft^2/sec)
MUBAR	dynamic viscosity based on reference temperature, (N)(sec)/m^2 ((lbf)(sec)/ft^2)

Output corresponding to KGRAD contains the three gradients of velocity and Mach number along the surface computed by the spline curve-fit method:

DUDS dUE/dx, sec^{-1}

DMDS dME/dx, m^{-1} (ft^{-1})

DMDL dME/dSOL

Output corresponding to KSDE contains the numerical solution of the laminar and turbulent differential boundary-layer equations. In the laminar case, the solution is the correlation number (CORLN). In the turbulent case, the solution is the incompressible form factor (FORMI), and a function (F) of the momentum thickness. These solutions are printed with respect to the surface length (S).

Output corresponding to KLAM contains the variables used in the laminar subroutine to check for the position of instability and transition. RTHI, increasing from station to station, and RCRIT and RTRAN, decreasing from station to station, are used in this analysis. When RTHI grows larger than RCRIT, instability has occurred. When RTHI bypasses RTRAN, transition is assumed to occur. For most cases of supersonic nozzle flow, the flow is laminar. The variables listed are

CORLN correlation number

SHEAR shear parameter

DTH ratio of transformed displacement thickness to transformed momentum thickness

FORMTR transformed form factor

SHAPL Pohlhausen shape factor based on boundary-layer thickness

RTHI incompressible momentum-thickness Reynolds number

SHAPK dimensionless shape factor based on momentum thickness

RCRIT critical incompressible momentum-thickness Reynolds number

KBAR mean shape factor based on momentum thickness

DIFF difference between transition and instability momentum-thickness Reynolds numbers

RTRAN incompressible momentum-thickness Reynolds number used in checking for transition point

Output corresponding to KPROF contains the velocity profiles at each station along the surface. The output listed is

Y/DELTA ratio of distance normal to surface in y-direction in boundary-layer profile to boundary-layer thickness

Y	distance normal to surface in y-direction in boundary-layer profile, m (ft)
Y/XMAX	ratio of Y to maximum X-coordinate
U	velocity within boundary layer, m/sec (ft/sec)
U/UE	ratio of U to free-stream velocity

REFERENCES

1. Vanco, Michael R.; and Goldman, Louis J.: Computer Program for Design of Two-Dimensional Supersonic Nozzle with Sharp-Edged Throat. NASA TM X-1502, 1968.
2. Goldman, Louis J.; and Scullin, Vincent J.: Analytical Investigation of Supersonic Turbomachinery Blading. I - Computer Program for Blading Design. NASA TN D-4421, 1968.
3. McNally, William D.: Fortran Program for Calculating Compressible Laminar and Turbulent Boundary Layers in Arbitrary Pressure Gradients. NASA TN D-5681, 1970.
4. Stewart, Warner L.: Analysis of Two-Dimensional Compressible-Flow Loss Characteristics Downstream of Turbomachine Blade Rows in Terms of Basic Boundary-Layer Characteristics. NACA TN 3515, 1955.
5. Cohen, Clarence B.; and Reshotko, Eli: The Compressible Laminar Boundary Layer with Heat Transfer and Arbitrary Pressure Gradient. NACA Rep. 1294, 1956.
6. Sasman, Philip K.; and Cresci, Robert J.: Compressible Turbulent Boundary Layer with Pressure Gradient and Heat Transfer. AIAA J., vol. 4, no. 1, Jan. 1966, pp. 19-25.
7. Goldman, Louis J.; and Vanco, Michael R.: Analytical Investigation of Two-Dimensional Loss Characteristics of Supersonic Turbine Stator Blades. NASA TM X-1823, 1969.

NASA-Langley, 1971 —— 12 E-6261

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
WASHINGTON, D.C. 20546

OFFICIAL BUSINESS
PENALTY FOR PRIVATE USE $300

FIRST CLASS MAIL

POSTAGE AND FEES PAID
NATIONAL AERONAUTICS AN
SPACE ADMINISTRATION

POSTMASTER: If Undeliverable (Section 158
Postal Manual) Do Not Retur

"The aeronautical and space activities of the United States shall be conducted so as to contribute . . . to the expansion of human knowledge of phenomena in the atmosphere and space. The Administration shall provide for the widest practicable and appropriate dissemination of information concerning its activities and the results thereof."

—NATIONAL AERONAUTICS AND SPACE ACT OF 1958

NASA SCIENTIFIC AND TECHNICAL PUBLICATIONS

TECHNICAL REPORTS: Scientific and technical information considered important, complete, and a lasting contribution to existing knowledge.

TECHNICAL NOTES: Information less broad in scope but nevertheless of importance as a contribution to existing knowledge.

TECHNICAL MEMORANDUMS: Information receiving limited distribution because of preliminary data, security classification, or other reasons.

CONTRACTOR REPORTS: Scientific and technical information generated under a NASA contract or grant and considered an important contribution to existing knowledge.

TECHNICAL TRANSLATIONS: Information published in a foreign language considered to merit NASA distribution in English.

SPECIAL PUBLICATIONS: Information derived from or of value to NASA activities. Publications include conference proceedings, monographs, data compilations, handbooks, sourcebooks, and special bibliographies.

TECHNOLOGY UTILIZATION PUBLICATIONS: Information on technology used by NASA that may be of particular interest in commercial and other non-aerospace applications. Publications include Tech Briefs, Technology Utilization Reports and Technology Surveys.

Details on the availability of these publications may be obtained from:

SCIENTIFIC AND TECHNICAL INFORMATION OFFICE

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
Washington, D.C. 20546